

Các công trình nghiên cứu, phát triển và ứng dụng
Công nghệ Thông tin và Truyền thông

<https://www.ictmag.vn/cntt-tt>

Tập 2024, Số 3, Tháng 12
Số đặc biệt TN-CT 2024

MỤC LỤC

A Large Language Model-Based Question Answering System for Online Public Administrative Services	
..... <i>Dinh-Dien La, Tuan-Anh Nguyen, Duc-Huy Mai, Tuan-Anh Nguyen, Thi-Thanh Ha, Trung-Nghia Phung, Van-Khanh Tran</i>	119
Evolutionary Algorithm for the Earliness-Tardiness Project Scheduling Problem	
..... <i>Hao Nguyen Thi, Huu Dang Quoc, Loc Nguyen The</i>	128
Phi-3 Meets Law: Fine-tuning Mini Language Models for Legal Document Understanding	
..... <i>Nguyen Huu Khanh, Nguyen Van Viet, Nguyen The Vinh, Nguyen Huu Cong</i>	136
Cold Storage Temperature Monitoring with Trilinear Interpolation	
..... <i>Minh Thai Truong, Minh Bang Le, Cong Danh Cao, Phuong Lan Phan, Huynh Tram Vo</i>	143
An Extended Algorithm using a Divide-and-Rule Approach for Boundary Value Analysis and Equivalence Class Partitioning	
..... <i>Danh Nguyen-Cong</i>	151
Utilizing Blockchain Technology to Secure Database from Unauthorized Modifications	
..... <i>Cu Vinh Loc, Tran Thi Dinh Xuyen, Truong Xuan Viet, Tran Hoang Viet, Le Hoang Thao, Truong Minh Thai</i>	159
Integrating Self-Supervised Learning with Nonlinear Classifiers in Lightweight Swin Transformer for X-Ray Image Classification	
..... <i>Tri-Thuc Vo, Thanh-Nghi Do</i>	169
Rice Leaf Disease Classification Using Knowledge Distillation	
..... <i>Md. Saiful Niaz, Tasmia Hassan Shabonty, Ashfaqur Rahman Saad, Lucky Talukder Tarin, Dewan Md. Farid, and Huu-Hoa Nguyen</i>	178
A Deep Learning based Approach for Student Attendance Recognition	
..... <i>Minh Nguyet Nguyen, Phuong Lan Phan</i>	187

Các công trình nghiên cứu, phát triển và ứng dụng
Công nghệ Thông tin và Truyền thông

<https://www.ictmag.vn/cntt-tt>

Tập 2024, Số 3, Tháng 12
Số đặc biệt TN-CT 2024

MỤC LỤC

Hệ thống hỏi đáp dịch vụ công trực tuyến dựa trên mô hình ngôn ngữ lớn.....
..... **Dinh-Dien La, Tuan-Anh Nguyen, Duc-Huy Mai, Tuan-Anh Nguyen, Thi-Thanh Ha, Trung-Nghia Phung, Van-Khanh Tran** 119

Thuật toán tiền hóa cho bài toán lập lịch dự án Earliness-Tardiness
..... **Hao Nguyen Thi, Huu Dang Quoc, Loc Nguyen The** 128

Phi-3 với Luật pháp: Tinh chỉnh mô hình ngôn ngữ nhỏ trong việc hiểu văn bản luật
..... **Nguyen Huu Khanh, Nguyen Van Viet, Nguyen The Vinh, Nguyen Huu Cong** 136

Giám sát nhiệt độ kho lạnh bằng phương pháp nội suy ba chiều.....
..... **Minh Thai Truong, Minh Bang Le, Cong Danh Cao, Phuong Lan Phan, Huynh Tram Vo** 143

Một thuật toán mở rộng sử dụng phương pháp chia để trị cho kỹ thuật phân tích giá trị biên và phân vùng tương đương
..... **Danh Nguyen-Cong** 151

Giám sát nhiệt độ kho lạnh bằng phương pháp nội suy ba chiều.....
..... **Cu Vinh Loc, Tran Thi Dinh Xuyen, Truong Xuan Viet, Tran Hoang Viet, Le Hoang Thao, Truong Minh Thai** 159

Tích hợp học tự giám sát với thuật toán phân lớp phi tuyến trong kiến trúc Lightweight Swin Transformer để phân lớp hình ảnh X-quang
..... **Tri-Thuc Vo, Thanh-Nghi Do** 169

Phân loại bệnh lá lúa bằng phương pháp chưng cất kiến thức
..... **Md. Saiful Niaz, Tasmia Hassan Shabonty, Ashfaqur Rahman Saad, Lucky Talukder Tarin, Dewan Md. Farid, and Huu-Hoa Nguyen** 178

Điểm danh sinh viên theo tiếp cận dựa trên mạng học sâu
..... **Minh Nguyet Nguyen, Phuong Lan Phan** 187

A Large Language Model-Based Question Answering System for Online Public Administrative Services

Dinh-Dien La¹, Tuan-Anh Nguyen², Duc-Huy Mai^{2†}, Tuan-Anh Nguyen^{2‡}, Thi-Thanh Ha², Trung-Nghia Phung², and Van-Khanh Tran²

¹ Department of Information and Communications, Ha Giang Province, Viet Nam.

² Institute of Applied Science and Technology - IAST, University of Information and Communication Technology, Thai Nguyen University, Vietnam.

Correspondence: Van-Khanh Tran, tvkhanh@ictu.edu.vn

Received: 02/07/2024, revised: 03/09/2024, accepted: 13/09/2024

Digital Object Identifier: 10.32913/mic-ict-research-vn.v2024.n3.1290

Abstract: Public administrative services are one of the key concerns for citizens; however, there are still numerous procedures and issues that citizens encounter and need clarification on. This study aims to alleviate some of this burden on officials by developing a Question Answering System (QnA) to address inquiries about public administrative services. We present a framework for utilizing Large Language Models to develop and implement a QnA system on the Online Public Administrative Service Portal in Ha Giang Province. We also provide extensive experimental analysis to demonstrate the effectiveness of each component in the QnA system.

Keywords: *Question answering system, public administrative services, large language model*

Tiêu đề: Hệ thống hỏi đáp dịch vụ công trực tuyến dựa trên mô hình ngôn ngữ lớn

Tóm tắt: Dịch vụ hành chính công là một trong những mối quan tâm chính của người dân, tuy nhiên vẫn còn nhiều thủ tục và vấn đề mà người dân gặp phải và cần được giải đáp. Nghiên cứu này nhằm giảm bớt một phần gánh nặng cho các cán bộ bằng cách phát triển Hệ thống Hỏi đáp để giải quyết các thắc mắc về dịch vụ hành chính công. Chúng tôi trình bày một nền tảng sử dụng các mô hình ngôn ngữ lớn nhằm phát triển và triển khai hệ thống Hỏi đáp trên cổng Dịch vụ hành chính công trực tuyến tại tỉnh Hà Giang. Chúng tôi cũng cung cấp phân tích thực nghiệm chuyên sâu để chứng minh tính hiệu quả của từng thành phần trong hệ thống Hỏi đáp.

Từ khóa: *Question answering system, public administrative services, large language model*

I. INTRODUCTION

Operating with a commitment to transparency and accessibility, the National Public Service Portal¹ serves as a centralized platform dedicated to assisting individuals and businesses, connecting and providing information on administrative procedures and online public services. It supports the implementation, monitoring, and evaluation of administrative procedure resolution, and online public services, as well as receiving, and handling feedback, and suggestions from individuals and organizations nationwide. The most common bottleneck in the resolution of administrative procedures often lies in the documentation guidance

and preparation. Many applications are redundant or lack necessary documentation as per regulations, leading to submissions to the wrong authority for resolution. If this stage is effectively managed, the time required to complete administrative procedures following the law would be significantly reduced, alleviating the burden on both citizens and the officials responsible for receiving and processing applications. In addition, the primary channels for information exchange between the government and citizens are through direct interactions, telephone calls, or social networks. This often results in delays for citizens and repetitive responses from officials.

To provide additional interactive channels between individuals, organizations, and the public services offered

[†]First batch students in IAST Young Talent Program, 2024

¹<https://dichvucong.gov.vn>

by the National Public Service Portal in a quick, timely, 24/7 manner, with personalized 1-1 assistance, and to enhance satisfaction among citizens and businesses with transparency, an artificial intelligence assistant system (a.k.a. QnA [1], or Chatbot [2]) needs to be implemented. This QnA system facilitates information retrieval from the service portal and automatically answers user queries related to many areas such as legal document searches, online public services, and administrative procedures.

Large Language Models (LLMs) [3] represent the cutting edge of natural language processing [4]. The evolution of LLMs [3] has demonstrated promising opportunities for enhancing the efficiency and accessibility of online public administrative services. To the best of our knowledge, this paper is the first work that presents a detailed experimental analysis of LLMs' application in online public administrative services, with a focus on QnA's capabilities in Ha Giang province², Vietnam. Our main contributions are threefold:

- Development of public administration QnA datasets: We present a comprehensive set of QnA datasets that cover a wide range of queries related to public administrative services in the National public service portal, Ha Giang portal, and three other provinces' portal.
- Framework for implementing QnA system: We propose a framework for integrating LLMs into QnA system for public administrative services.
- Experimental analysis for each stage in the framework: We provide extensive experiments and evaluations for the QnA system.

II. RELATED WORKS

QnA systems in public administration are designed to handle citizen inquiries and provide timely, accurate information. Several cities in Vietnam are exploring the integration of LLM technology, i.e., ChatGPT³ into their public administration systems to enhance service delivery and efficiency, such as Ho Chi Minh, Hanoi, Quang Nam, and Da Nang, etc. Authors in [5] present a model to retrieve public administrative information and answer questions related to Vietnamese legal documents. To our knowledge, this is the first work in the literature that provides detailed descriptions of public administrative processes and presents a systematic approach to QnA systems in this context.

a) Online Public Service Portal in Ha Giang

To date, the Ha Giang province Online Public Service Portal has registered 33,434 accounts and received more

than 399,440 visits this year. On average, the portal handles approximately 950 applications daily, with an online submission rate of around 90%. This work provides a comprehensive dataset including a wide range of public administrative questions and answers, as well as detailed information on administrative procedures.

b) Large Language Models

LLMs leverage deep learning techniques and massive computational resources to process and generate language, making them capable of a wide range of tasks, from simple question answering to complex text generation and language translation. The rapid advancement of LLMs has opened new avenues for enhancing public administrative services, particularly in the context of online platforms. However, research publications on the application of LLMs in the context of Vietnam's public administrative services remain underexplored. In this work, we experiment with some Vietnamese pre-trained LLMs such as Vinallama-7b-chat⁴, Vistral-7b-chat⁵, and phoGPT-4b-chat⁶, running on local devices. Using Vietnamese Large Language Models, which have been further pre-trained on a substantial volume of relevant Vietnamese data, significantly enhances their ability to understand and process specific laws, regulations, and public services. This leads to better performance in retrieval, reasoning, and answer generation specific to the Vietnamese context. Furthermore, we fine-tune the model using the LoRA method [6] on the training dataset to enhance model's capability to better adapt to the public administration domain.

III. METHODS

1. Datasets

a) QnA Dataset

We provide a comprehensive collection of Q&A datasets covering a wide array of queries related to online public administrative services. These datasets are sourced from the Ha Giang Portal with 1,627 data samples, the National Public Service Portal and additional portals for the provinces of Bac Ninh, Quang Ninh, and Bac Giang in total of 11,334 QnA pairs. We manually selected 202 QnA samples to serve as the test set. Question types of the dataset are in seven categories, including Reasoning Questions, Factoid Questions, Yes/No Questions, Multiple-choice Questions, and Questions involving multiple agencies, multiple relevant documents, and multiple relevant articles. There is a mix of question types, including reasoning-based, factual, yes/no,

²<https://dichvucong.hagiang.gov.vn>

³<https://openai.com/>

⁴<https://huggingface.co/vilm/vinallama-7b-chat>

⁵<https://huggingface.co/Viet-Mistral/Vistral-7B-Chat>

⁶<https://huggingface.co/vinai/PhoGPT-4B-Chat>

and multiple-choice questions, reflecting a diverse set of queries that may require different approaches to answer.

b) Administrative Procedure Dataset in Ha Giang

In compliance with the directives from the Chairman of the People's Committee of Ha Giang Province, we have compiled a comprehensive dataset of 1,943 administrative procedures⁷. Each entry in this dataset includes the following details: ID, link, name, status, sequence of execution, target subjects, methods of implementation, required documents, expected outcomes, resolution time, and the responsible agency, among other relevant information.

The longest procedure in our dataset contains 7,490 words, while the shortest has only 132 words. Due to this variability, some procedures exceed the input limits of embedding and retrieval models. This significant range in word counts underscores the necessity for thorough pre-processing and segmentation to ensure consistent model performance across varying document sizes. Additionally, the third quartile (Q3) value indicates that most samples are considerably shorter than the maximum length, highlighting the presence of a few outliers with exceptionally high word counts. On average, questions in the dataset are about 36 words long and feature numerous key vocabulary terms that enhance retrieval effectiveness. Table III shows statistics of datasets.

By compiling the areas that frequently receive inquiries from citizens to the 1022⁸ hotline of the province, we identified several priority areas for analysis of the questions, including border gate economic zone management, industry and trade, education, and training, among others. Corresponding to these priority areas, we manually extracted 202 QnA pairs from a total of 1,943 administrative procedures. Table I and II summarize a classification of 202 questions from a test set based on different categories, including:

- Reasoning Questions (RQ): These are questions that require logical reasoning to answer, with a total of 14 questions in this category.
- Factoid Questions (FQ): Factoid questions are those that can be answered with specific, extractable information from texts, such as factual data or details from documents. There are 173 such questions, which make up the majority of the dataset.
- Yes/No Questions (YN): These are simple questions where the answer is either "yes" or "no." There are 14 questions in this category.
- Multiple Choice Questions (MC): These questions provide multiple answer options, from which the correct

answer must be chosen. There are 20 multiple-choice questions.

- Number of Relevant Documents (ND): This represents the number of legal documents related to each query. There are 187 queries related to 206 legal documents.
- Number of Relevant Articles (NA): This refers to the number of legal articles relevant to each query. There are 92 queries related to 138 legal articles.
- Questions Involving Multiple Agencies (MA): These questions require input or resolution from multiple government agencies. There are 57 such questions.

While Factoid Questions (FQ) dominate the dataset, comprising 85.64% of the total questions, a significant number of questions (187 out of 202) relate to legal documents, and 92 of them specifically relate to articles within those documents. This classification highlights the heavy emphasis on fact-based questions and legal document references, suggesting that the dataset may be designed for scenarios where retrieving and applying legal or procedural information is critical.

c) Administrative Documents Segmentation

The deep learning model follows input constraints by dividing administrative documents into smaller segments aligned with specific administrative procedure fields. This segmentation strategy maintains semantic coherence within each document segment, ensuring the integrity of information retrieval for improved accuracy and contextual relevance of outcomes. Moreover, administrative procedure identifiers are prefixed to segmented documents, providing essential context, especially for concise fields such as fees, charges, agency levels, and other specifics. Figure 1 depicts the length distribution of administrative data before and after segmentation.

2. Administrative Procedure Retrieval

Retrieving documents based on a query involves searching for and selecting appropriate documents to fulfill specific information needs. This typically includes employing models and algorithms to assess the relevance of documents to the query. BM25 [7] relies solely on keyword matching and word frequency within documents, without capturing semantic relationships between words. While it performs well with straightforward queries and clear keywords, BM25 struggles with complex queries that involve multiple layers of meaning or contextual nuances, potentially leading to the omission of relevant documents. Additionally, BM25 may be less effective with documents that have irregular styles, spelling errors, or word variations, unless supplemented by additional preprocessing techniques.

⁷<https://dichvucong.hagiang.gov.vn/vi/procedure/search?keyword=>

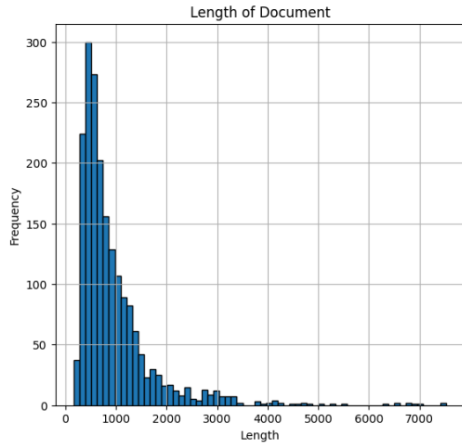
⁸<https://1022.hagiang.gov.vn/vi/>

Table I
STATISTICS OF QUESTION TYPES

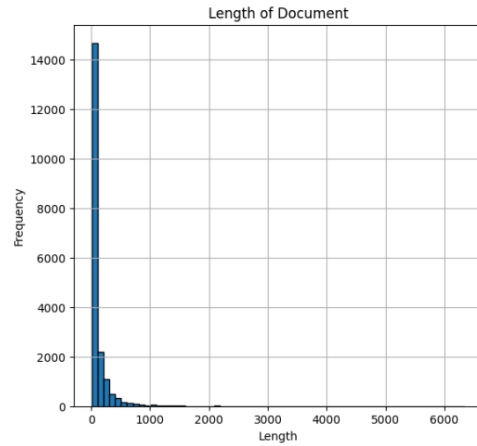
No.	Question Type	Symbol	Quantity
1	Reasoning Questions: Questions involving reasoning	RQ	14
2	Factoid Questions: Questions with extractable answers	FQ	173
3	Yes/No Questions	YN	14
4	Multiple Choice Questions	MC	20
5	Number of relevant documents for each query	ND	187
6	Number of relevant articles for each query	NA	92
7	Questions involving multiple agencies	MA	57

Table II
QUESTION CLASSIFICATION BY GROUP

No.	Question Group	Quantity
1	Factoid Questions (FQ)	112
2	Factoid Questions involving multiple agencies (FQ-MA)	45
3	Factoid-Multiple Choice-Multiple Agencies (FQ-MC-MA)	11
4	Factoid-Multiple Choice Questions (FQ-MC)	3
5	Factoid-Yes/No Questions (FQ-YN)	1
6	Factoid Questions-Multiple Agencies (FQ-MA)	1
7	Multiple Choice Questions (MC)	5
8	Reasoning Questions (RQ)	10
9	Reasoning-Multiple Choice Questions (RQ-MC)	1
10	Reasoning-Yes/No Questions (RQ-YN)	3
11	Yes/No Questions (YN)	10



(a) before pre-processing.



(b) after pre-processing.

Figure 1. Document length (a) before and (b) after pre-processing

Table III
STATISTICS OF ADMINISTRATIVE PUBLIC DATASET

File	Count	Words per sample			
		Min	Max	Avg	Q3
No. of administrative procedures	1,943	132	7490	911	1081
Question (train)	11,334	5	464	30	36
Answer (train)	11,334	1	4,909	174	218
Question (test)	202	7	129	36	41
Answer (test)	202	1	979	131	163

On the other hand, BERT's [8] performance heavily depends on the quality of its training data and fine-tuning.

If the data is not representative or properly tailored to the application context, BERT may yield inaccurate results. While BERT processes sentences word by word, SBERT [9] considers entire sentences, providing a more comprehensive understanding of text semantics. This allows search engines to better interpret and categorize content, leading to more in-depth content indexing and more precise search results.

Combining BM25 and SBERT can create a more robust evaluation system by integrating semantic understanding with keyword matching. This is achieved by normalizing the scores from each method and aggregating them using an integrated formula. In this work, we set up an ensemble

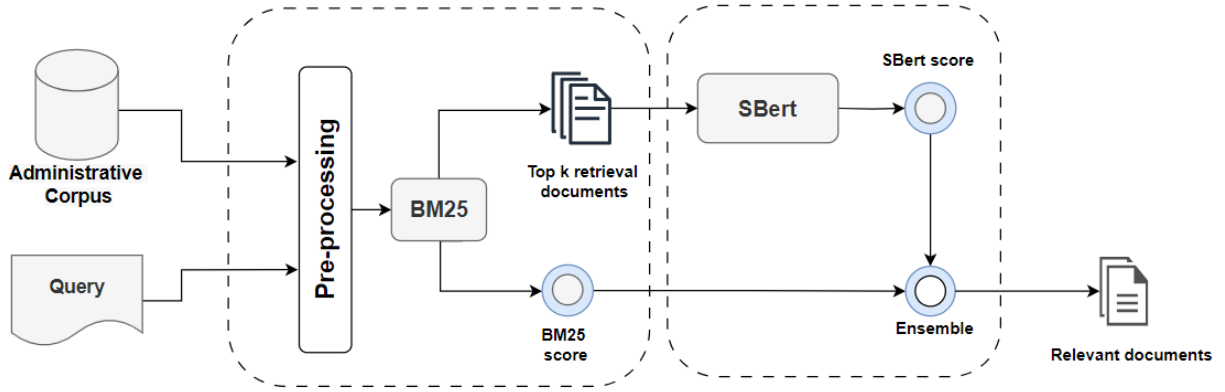


Figure 2. Architecture of the Administrative Procedure Retrieval

method that integrates lexical search BM25 with a semantic search SBERT model. Figure 2 shows the retrieval architecture while experimental results are presented in section IV.1.

a) Text Ranking

Keyword Ranking: The utilization of Okapi BM25 [7] for keyword ranking is an established lexicon-based approach aimed at prioritizing administrative documents. This algorithm assesses administrative procedure relevance by analyzing the frequency of vocabulary terms present in queries. By leveraging commonly used keywords and specialized terminology found in inquiries and related literature, BM25 enhances the model's ability to retrieve pertinent information while optimizing computational efficiency. However, its effectiveness may be constrained when confronted with queries containing low-value vocabulary or ambiguous content, such as "Chi phí đền bù" ("Compensation" in English).

Semantic Ranking: Semantic ranking focuses on assessing semantic relevance. The process entails breaking down documents into coherent segments or sentences, which are then transformed into multidimensional vectors. This transformation is also applied to the query, enabling ranking based on semantic similarity. We use Vietnamese-Bi-Encoder model [10], a specific SBERT model for the Vietnamese language, for this task and further fine-tune the model on public administrative datasets for enhancing embedding and semantic search.

b) Ensemble

At this stage, we use min-max scaling algorithm to normalize the output scores x for each query q as:

$$\text{norm}(x) = \frac{x - m_q + 0.01}{M_q - m_q + 0.01} \quad (1)$$

where $M_q = \max(\text{BM25})$ and $m_q = \min(\text{BM25})$. The constant 0.01 is added to prevent division by zero when $M_q = m_q$. The final score is computed as below:

$$\text{Ensemble-Score} = \alpha \cdot w_{\text{BM25}} + (1 - \alpha) \cdot w_{\text{SBERT}} \quad (2)$$

We then employ grid search to determine the optimal weight for parameter α , constrained within the range of 0 to 1.

3. Administrative Procedure Answer Generation

The process of generating answers from retrieved candidate articles aims to gather pertinent information that fulfills the user's information requirements efficiently. This involves identifying key passages within the articles and synthesizing them to construct a coherent and comprehensive response. After the text retrieval stage, the relevant documents undergo preprocessing before being fed into the large language model (LLM). Specifically, synthesize related documents one after another to form a document with enough information. The context within documents is improved by adding additional information from the corpus while removing information with low similarity scores. This process is illustrated in Figure 3.

a) Prompt

A prompt [11] is a piece of text or statement that provides context or specific requirements, and instructions for the model to understand and produce appropriate results. We use a simple prompt that asks the model to get the contextual information provided that the documents are retrieved to answer the question.

b) Choosing LLM for Generating Task

For a QnA task focused on Vietnamese, especially in specialized domains such as law and governance, a Vietnamese large language model offers significant advantages

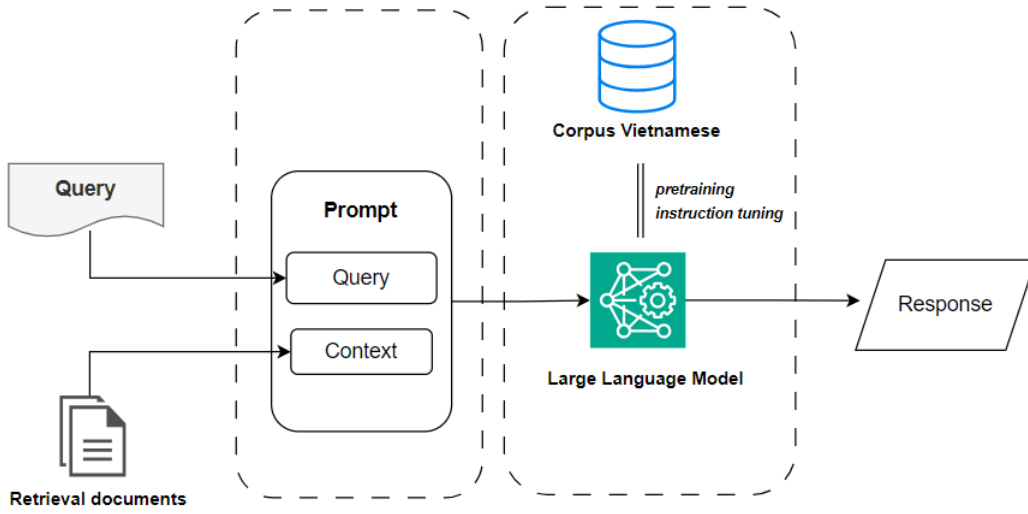


Figure 3. Architecture of the Administrative Procedure Answer Generation

in terms of language proficiency, domain specialization, performance efficiency, and alignment with local needs. Vietnamese LLM candidates to chose were Vinallama-7b-chat⁹, Vistral-7b-chat¹⁰, and phoGPT-4b-chat¹¹. By leveraging methods like LoRA for fine-tuning, the model can be further optimized to handle specific types of queries effectively, ensuring that it provides the most accurate and relevant responses for Vietnamese users.

c) Fine-tuning LLMs

LLMs demonstrated advanced performances in specific domains, but their effectiveness in the public administrative domain is unknown. This provides an opportunity to study whether our chosen LLMs (fine-tuned or not) have high performance on administrative document understanding. The fine-tuning process for LLMs involves several steps to optimize them for specific applications. First, a Vietnamese LLM, such as Vistral 7b chat, is selected and reduced in size using 4-bit quantization, which decreases memory requirements while maintaining high performance. Next, a training dataset consisting of 11,334 QnA pairs about public administrative procedures is prepared. This dataset is formatted by adding special tokens like <s>, </s>, [INST], and [/INST] for training purposes. The model is then fine-tuned using this data, with a specific LoRA configuration designed to enhance processing efficiency and reduce computational costs.

During the fine-tuning process, an A10G GPU with

24GB of RAM was utilized, completing the process in approximately 8 hours over 3 epochs, for a total of 4,249 steps. The LoRA configuration included a lora_alpha of 16, a dropout rate of 0.1, and a rank of 64, targeting specific modules such as q_proj, k_proj, v_proj, o_proj, and gate_proj. The training setup featured a per-device batch size of 8, gradient accumulation steps set to 8, and the paged_adamw_32bit optimizer, ensuring efficient processing throughout the tuning process.

d) Evaluating Answer Generation Task

To evaluate the quality of answer generation for administrative procedures and the performance of our QnA system (detailed in Section IV.3), we employ RAGAS (Retrieval Augmented Generation Assessment) as outlined by [12]. RAGAS is a comprehensive framework for assessing Retrieval Augmented Generation (RAG) systems, offering a variety of useful metrics for thorough evaluation. The main RAGAS metrics for this task are Faithfulness, Answer Similarity, Answer Correctness, and Answer Relevancy, details as below:

- Faithfulness measures the factual consistency of the generated response against the retrieved context: if all the claims that are made in the answer can be inferred from the given context then the response is considered “faithful” to the provided context.
- Answer similarity measures the semantic resemblance between the generated answer and the ground truth answer. This is done via cosine similarity between the embedding vectors of the ground truth answer and the generated answer.

⁹<https://huggingface.co/vilm/vinallama-7b-chat>

¹⁰<https://huggingface.co/Viet-Mistral/Vistral-7B-Chat>

¹¹<https://huggingface.co/vinai/PhoGPT-4B-Chat>

- Answer relevancy measures how pertinent the generated question is to the given question. This is computed by generating a number of artificial questions based on the answer and measuring the similarity between the original question and those artificial questions.
- Answer correctness measures how accurate the generated answer is relative to a “golden” answer that is deemed to be the correct answer. It is based on a weighted sum of factual consistency and the semantic similarity between the ground-truth answer and the generated response.

Utilizing GPT-4 to assess the output of chosen LLMs within the RAGAS metrics framework allows for a more reliable and comprehensive evaluation of LLMs, ensuring they meet the standards required for specialized tasks, such as legal and procedural QnA in Vietnamese.

IV. EXPERIMENT RESULTS

1. Administrative Procedure Retrieval

In this task, we use precision, recall, and F2-macro [13] to assess retrieval performance. The metrics are computed as follows:

$$\begin{aligned} \text{Precision}_i &= \frac{\text{Number of correctly retrieved articles of question } i}{\text{Number of retrieved articles of question } i} \\ \text{Recall}_i &= \frac{\text{Number of correctly retrieved articles of question } i}{\text{Number of relevant articles of question } i} \\ F2_i &= \frac{5 \times \text{Precision}_i \times \text{Recall}_i}{4 \times \text{Precision}_i + \text{Recall}_i}; \text{ and } F2 = \text{Average of } (F2_i) \end{aligned}$$

Pre-processed data is first fed to the BM25 component and top-k documents with best BM25 scores are retrieved. Table V shows recall performances for a variety of chosen top-k. As expected, the recall is highest for larger top-k values. For $k = 500$, the recall reaches 99%, indicating that nearly all relevant documents are retrieved when considering a large set of results. This shows that BM25 is effective at retrieving relevant documents when allowed to return a large number of candidates. In our experiments, we select 100 articles with the highest BM25 score for this task.

Combining lexical-based search BM25 with semantic search SBERT has demonstrated significantly better performance compared to using BM25 alone. In the ensemble stage (Section III.2.b), we use grid search to find the best weight α for BM25 and SBERT. The grid search results in table IV shows the F2-score for different values of α across various top-k settings. The highest F2-scores are observed for $\alpha = 0.3$, with the top scores across multiple top-k settings, particularly for $k = 3, 4, 5, 6$, where the F2-score reaches its peak at 0.7787 and 0.7793. This suggests that $\alpha = 0.3$ is the optimal choice for maximizing the F2-score.

Table VI presents the results of administrative procedure retrieval using two methods, BM25, and an ensemble approach, across different top-k values. BM25 performs reasonably well at lower top-k values but starts to decline as more results are included. Even though BM25 achieves relatively good recall at higher top-k values, its precision drops too much, impacting the overall F2-score, especially at $k = 10$. The ensemble method achieves better F2-scores across all top-k values, with the highest F2-score at $k = 6$ (0.7793). This indicates that the ensemble method is more balanced, giving greater emphasis to recall, which aligns with the F2-score’s focus on recall.

2. Administrative Procedure Answer Generation

We conducted an evaluation for this task using common metrics such as ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-Lsum, and BLEU. Results are shown in Table VII. Vinallama-7b-chat leads in most ROUGE metrics, particularly in unigram, bigram, and sequence overlap, indicating it generates answers closely resembling the reference texts in terms of word and sequence similarity, while Vistral-7b-chat performs best in BLEU score, suggesting it generates high-quality, fluent answers with good n-gram precision.

We utilized GPT-4 to evaluate the quality of answer generation for administrative procedures and the performance of our QnA system within the RAGAS metrics framework. Results are shown in Table VIII. Vistral-7b-chat leads with the highest correctness score (0.8009), followed by Vinallama-7b-chat (0.7648), and phoGPT-4b-chat lags behind (0.6341). This suggests that Vistral-7b-chat provides the most accurate answers, while phoGPT-4b-chat needs improvement in generating correct responses.

3. Administrative Procedure Question Answering

In this task, we use RAGAS[12] with 4 evaluation indicators for each component of the QnA system, such as Context Recall, Context Precision, Faithfulness, and Answer Relevance. For each question in the 202 questions in the test set, we retrieve top-k = 6 articles as input into the large language models to generate answers. Results are shown in Table IX.

Overall, Vistral-7b-chat outperforms the other models across all metrics since it has the highest context precision (0.76538), context recall (0.72835), faithfulness (0.7509), and answer relevance (0.72483). Faithfulness, which measures how accurately the answers reflect the information in the context, is highest in Vistral-7b-chat (0.7509), indicating its answers are the most reliable. The model also leads with the highest answer relevance (0.72483), showing that it provides the most pertinent and appropriate responses to questions.

Table IV
F2-SCORE RESULTS FOR DIFFERENT α AND TOP-K

α	$top - k$									
	1	2	3	4	5	6	7	8	9	10
1.0	0.6364	0.6512	0.6582	0.6334	0.6448	0.6420	0.6329	0.6319	0.6186	0.6029
0.9	0.6463	0.6644	0.6587	0.6651	0.6617	0.6551	0.6547	0.6437	0.6343	0.6219
0.8	0.6661	0.6719	0.6933	0.6931	0.6931	0.6859	0.6789	0.6662	0.6562	0.6463
0.7	0.7027	0.7181	0.7196	0.7139	0.7047	0.7045	0.6942	0.6827	0.6731	0.6616
0.6	0.7197	0.7628	0.7433	0.7405	0.7300	0.7241	0.7159	0.7095	0.6967	0.6810
0.5	0.7365	0.7546	0.7612	0.7577	0.7636	0.7590	0.7393	0.7256	0.7152	0.6951
0.4	0.7315	0.7628	0.7785	0.7781	0.7782	0.7785	0.7656	0.7446	0.7333	0.7132
0.3	0.7315	0.7735	0.7787	0.7787	0.7785	0.7793	0.7729	0.7543	0.7365	0.7184
0.2	0.7513	0.7710	0.7667	0.7673	0.7782	0.7783	0.7687	0.7500	0.7313	0.7180
0.1	0.7563	0.7660	0.7692	0.7673	0.7628	0.7620	0.7591	0.7406	0.7309	0.7133
0.0	0.7414	0.7503	0.7580	0.7639	0.7585	0.7550	0.7408	0.7335	0.7244	0.7075

Table V
RECALL PERFORMANCES FOR DIFFERENT TOP-K VALUES OF BM25.

Top k	500	100	50	10	3	1
Score %	99	97	92	83	75	63

Table VI
RESULTS OF ADMINISTRATIVE PROCEDURE RETRIEVAL

Method (top-k)	Precision	Recall	F2 score
BM25 (k=1)	0.6386	0.6386	0.6386
BM25 (k=2)	0.542	0.7037	0.6641
BM25 (k=3)	0.4991	0.7549	0.6847
BM25 (k=5)	0.4624	0.7871	0.6902
BM25 (k=10)	0.3809	0.83	0.6716
Ensemble (k=1)	0.7376	0.7310	0.7323
Ensemble (k=2)	0.7029	0.8102	0.7734
Ensemble (k=3)	0.674	0.8432	0.7787
Ensemble (k=4)	0.6592	0.8605	0.7784
Ensemble (k=5)	0.6481	0.8778	0.7784
Ensemble (k=6)	0.6367	0.8968	0.7793
Ensemble (k=7)	0.6178	0.9067	0.7729
Ensemble (k=8)	0.5785	0.9084	0.7543
Ensemble (k=9)	0.5355	0.915	0.7365
Ensemble (k=10)	0.4972	0.9199	0.7184

Table VII
RESULTS OF ANSWER GENERATION TASK

Model	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-Lsum	BLEU
Vinallama-7b-chat	0.5653	0.3910	0.4216	0.4889	0.2513
Vistral-7b-chat	0.546	0.3835	0.4087	0.4803	0.2533
phoGPT-4b-chat	0.4655	0.312	0.3511	0.404	0.1864

V. CONCLUSION

This study has demonstrated a QnA system for the Online Public Administrative Service Portal in Ha Giang Province, utilizing Large Language Models and contributing a comprehensive dataset that serves as a valuable resource for future research. Despite the potential of LLMs, their application in Vietnam's public administrative services remains

Table VIII
RESULTS OF ANSWER GENERATION TASK

Model	Answer similarity	Answer correctness	Faithfulness	Answer Relevance
Vinallama-7b-chat	0.828988	0.7648	0.743966	0.758991
Vistral-7b-chat	0.917168	0.8009	0.740286	0.726984
phoGPT-4b-chat	0.677705	0.6341	0.6802	0.836788

Table IX
THE RESULT OF END-TO-END QNA SYSTEM

Model	Context Precision	Context Recall	Faithfulness	Answer Relevance
Vinallama-7b-chat	0.72759	0.685632	0.67094	0.692839
Vistral-7b-chat	0.76538	0.72835	0.7509	0.72483
phoGPT-4b-chat	0.67919	0.64319	0.61489	0.56775

underexplored. Our study addresses this gap by offering a systematic approach to integrating LLMs into public administrative frameworks. However, further research is needed to explore the broader implications of LLMs in diverse administrative contexts, including approaches like advanced RAG [14], or autonomous Agents [15]. Additionally, it is crucial to effectively address different question types within the QnA system, as each type requires tailored strategies for optimal handling.

REFERENCES

- [1] J. Martinez-Gil, "A survey on legal question-answering systems," *Computer Science Review*, vol. 48, p. 100552, 2023.
- [2] G. Caldarini, S. F. Jaf, and K. McGarry, "A literature survey of recent advances in chatbots," *CoRR*, vol. abs/2201.06657, 2022. [Online]. Available: <https://arxiv.org/abs/2201.06657>
- [3] S. Minaee, T. Mikolov, N. Nikzad, M. Chenaghlu, R. Socher, X. Amatriain, and J. Gao, "Large language models: A survey," 2024.
- [4] A. Torfi, R. A. Shirvani, Y. Keneshloo, N. Tavaf, and E. A. Fox, "Natural language processing advancements by deep learning: A survey," 2021. [Online]. Available: <https://arxiv.org/abs/2003.01200>

- [5] A. Pham Duy and H. Le Thanh, "A question-answering system for vietnamese public administrative services," in *Proceedings of the 12th SoICT*, ser. SOICT '23, 2023.
- [6] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," *CoRR*, vol. abs/2106.09685, 2021.
- [7] S. Robertson and H. Zaragoza, "The probabilistic relevance framework: Bm25 and beyond," *FTIR*, vol. 3, pp. 333–389, 01 2009.
- [8] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2019. [Online]. Available: <https://arxiv.org/abs/1810.04805>
- [9] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," 2019. [Online]. Available: <https://arxiv.org/abs/1908.10084>
- [10] N. Q. Duc, L. H. Son, N. D. Nhan, N. D. N. Minh, L. T. Huong, and D. V. Sang, "Towards comprehensive vietnamese rag and llms," *arXiv preprint arXiv:2403.01616*, 2024.
- [11] X. Amatriain, "Prompt design and engineering: Introduction and advanced methods," *arXiv preprint arXiv:2401.14423*, 2024.
- [12] S. Es, J. James, L. Espinosa-Anke, and S. Schockaert, "Ragas: Automated evaluation of retrieval augmented generation," *arXiv preprint arXiv:2309.15217*, 2023.
- [13] C. Nguyen, S. Luu, T. Tran, A. Trieu, A. Dang, D. Nguyen, H. Nguyen, T. Pham, T. Pham, T.-T. Vo, D.-T. Dol, N. Khang, H. Nguyen, N.-C. Le, T.-T. Le, Q. Bui, P. Nguyen, H.-T. Nguyen, V. Tran, and L. Nguyen, "A summary of the alqac 2023 competition," 10 2023, pp. 1–6.
- [14] M. Eibich, S. Nagpal, and A. Fred-Ojala, "Aragog: Advanced rag output grading," 2024. [Online]. Available: <https://arxiv.org/abs/2404.01037>
- [15] L. Wang, C. Ma, X. Feng, Z. Zhang, H. Yang, J. Zhang, Z. Chen, J. Tang, X. Chen, Y. Lin, W. X. Zhao, Z. Wei, and J. Wen, "A survey on llm based autonomous agents," *Frontiers of Computer Science*, vol. 18, no. 6, Mar. 2024.



Dinh-Dien La is a PhD student majoring in computer science, University of Information and Communications Technology, Thai Nguyen University. He is currently Deputy Director of the Department of Information and Communications of Ha Giang province. His research interests are data science, machine learning, and deep

learning.

Email: ladien.it@gmail.com



Tuan-Anh Nguyen is pursuing an Engineering degree in Information Technology at the University of Information and Communication Technology in Thai Nguyen, Vietnam. He is involved with the Institute of Applied Science and Technology in Thai Nguyen. His research interests include machine learning, deep learning, and natural

language processing.

Email: dtc20h4802010242@ictu.edu.vn



Duc-Huy Mai is pursuing the B.Eng degree in Information Technology from Thai Nguyen University of Information and Communication Technology (ICTU). His research interests include software development, natural language processing (NLP), and question answering.

Email: dtc1954802010132@ictu.edu.vn



Thi-Thanh Ha received PhD degree in Information System in Ha Noi University of Science and Technology, Viet Nam. She obtained Bachelor Degree of Science in Applied Mathematics and Informatics from University of Natural Science, Vietnam National University in 2004. She currently is a lecturer at Computer Science in Thai Nguyen University of Information and Communication Technology. Her research interests are fields of deep learning in Natural Language Processing, Question Answering, and chatbot.

Email: httthanh@ictu.edu.vn



Assoc. Prof. Trung-Nghia Phung received his Engineering degree in Electronics and Telecommunications from Hanoi University of Science and Technology (HUST) in 2002. He completed his Master of Science degree in Telecommunications from Vietnam National University –Hanoi (VNUH) in 2007 and his PhD degree in Information Science from Japan Advanced Institute of Science and Technology (JAIST) in 2013. He has been Rector of Thai Nguyen University of Information and Communication Technology (ICTU). His main research interests are signal processing and machine learning.

Email: ptnghia@ictu.edu.vn



Van-Khanh Tran received Ph.D. in Natural Language Processing from the Japan Advanced Institute of Science and Technology (JAIST), where his research focused on deep learning for natural language generation in spoken dialogue systems. contributed to the development of NLP applications. He is currently an AI Research Scientist on the NLP team at FPT Smart Cloud's Generative AI (GenAI) Center, where he focuses on developing large language models and AI assistant ecosystems tailored for Vietnamese users. He also serves as the Deputy Head of the Institute of Applied Science and Technology. His research interests include natural language processing, large language models, and AI applications in the legal, healthcare, and finance domains.

Email: tvkhanh@ictu.edu.vn

Evolutionary Algorithm for the Earliness-Tardiness Project Scheduling Problem

Hao Nguyen Thi¹, Huu Dang Quoc², Loc Nguyen The³

¹ Hung Vuong University, Nong Trang, Viet Tri, Phu Tho

² Thuong Mai University, 79 Ho Tung Mau, Cau Giay, Ha Noi, Viet Nam

³ Hanoi National University of Education

Correspondence: Huu Dang Quoc, huudq@tmu.edu.vn

Received: 29/06/2024, revised: 31/08/2024, accepted: 13/09/2024

Digital Object Identifier: 10.32913/mic-ict-research-vn.v2024.n3.1288

Abstract: Completing too early or too late can cause damage to the project. For example, in transporting goods to the seaports, if the shipping arrives too early, it will take time and cost to stay at the dock. In the other case, shipping too late will reduce the quality of the product and break the contract. The goal of this paper is to minimize the tardiness and earliness of those projects. First, some related studies are briefly introduced, thereby showing the role and practical applications of the MS-RCPSP (Multi-Skill Resource Constrained Project Scheduling Problem) problem and the two factors of earliness and delay in scheduling problems. The roles of earliness and tardiness in scheduling problems are also described. At the same time, the new contribution of this research is also highlighted, which is the proposal for the latest problem E-RCPSP (Earliness tardiness MS-RCPSP). This problem is the first problem in the MS-RCPSP family that considers the sum of earliness and tardiness as the objective, which has never been considered in previous studies. Then, the mathematical model of the proposed E-RCPSP problem, which includes the objective function, project components and constraints is introduced. To solve the proposed problem, two approximate algorithms based on the evolutionary algorithms GA (Genetic Algorithm) and DE (Differential Evolution) are introduced. In the last section of the article, the performance of these two algorithms is verified and compared with each other through experiments conducted on the iMOPSE benchmark dataset.

Keywords: *Network resources management, earliness-tardiness cost, resource-constrained project scheduling, evolutionary algorithms, differential evolution algorithms.*

Tiêu đề: Thuật toán tiến hóa cho bài toán lập lịch dự án Earliness-Tardiness

Tóm tắt: Hoàn thành quá sớm hoặc quá muộn có thể gây thiệt hại cho dự án. Ví dụ, trong việc vận chuyển hàng hóa đến cảng biển, nếu hàng hóa đến quá sớm, sẽ mất thời gian và chi phí để lưu lại tại bến tàu. Ở trường hợp khác, việc vận chuyển quá muộn sẽ làm giảm chất lượng sản phẩm và phá vỡ hợp đồng. Mục tiêu của bài báo này là giảm thiểu tình trạng các dự án được hoàn thành quá trễ hay quá sớm. Đầu tiên, bài báo giới thiệu tóm tắt một số nghiên cứu liên quan, qua đó chỉ ra vai trò và ứng dụng thực tế của bài toán MS-RCPSP (Multi-Skill Resource Constrained Project Scheduling Problem) và hai yếu tố Độ sớm và Độ trễ trong bài toán lập lịch. Đồng thời, tác giả nêu bật đóng góp mới của nghiên cứu này, đó là đề xuất một bài toán mới tên là E-RCPSP (Earliness tardiness MS-RCPSP). Bài toán mới này là bài toán đầu tiên trong họ MS-RCPSP đặt tổng Độ sớm và Độ trễ thành hàm mục tiêu, điều chưa từng xuất hiện trong các nghiên cứu trước đây về MS-RCPSP. Sau đó, bài báo giới thiệu mô hình toán học của bài toán E-RCPSP bao gồm hàm mục tiêu, các thành phần của dự án và các ràng buộc, đặc biệt là ràng buộc về thời gian. Để giải quyết bài toán E-RCPSP, bài báo xây dựng hai thuật toán gần đúng dựa trên thuật toán tiến hóa GA (Genetic Algorithm) và thuật toán tiến hóa DE (Differential Evolution). Phần cuối của bài báo trình bày những thực nghiệm trên tập dữ liệu chuẩn iMOPSE nhằm kiểm chứng và so sánh hiệu năng của hai thuật toán đề xuất.

Từ khóa: *Network resources management, earliness-tardiness cost, resource-constrained project scheduling, evolutionary algorithms, differential evolution algorithms.*

I. INTRODUCTION

The project schedule is the deciding factor in project efficiency, but finding the optimal schedule is very difficult

because of constraints on task priority and resource limitations. For that reason, MS-RCPSP (Multi-Skill Resource Constrained Project Scheduling Problem) [1], one of the typical project scheduling problems, has long been proven

to be an NP-Hard problem. This is a scheduling problem with many practical applications, so it has been of interest to many researchers [2], [3], [4], [5].

The MS-RCPSP scheduling problem assumes that the project is conducted by a given set of resources, each with a different skill level. A given project consists of tasks that are related to each other in order of priority, meaning that only after the parent task is completed can the child task begin to be processed. For a resource to perform a task, it must have the skill required by the task at a skill level greater than or equal to the level required by the task.

This problem was soon proven to be NP-Hard [1], meaning that its optimal solution cannot be found in polynomial time. Therefore, the practical approach often used by researchers is to find an approximate solution with an acceptable deviation.

Myszkowski and colleagues proposed the Ant colony algorithm [3] and GA algorithm [5] to solve the MS-RCPSP problem. In addition, a very important contribution of Myszkowski's research group is the construction of the iMOPSE dataset [2] which is considered a standard dataset and widely used by many other studies.

However, MS-RCPSP has limitations that prevent this problem from fully representing real projects. The limitation is that this problem aims to minimize makespan and cost without taking into account two other important factors that determine project efficiency: tardiness and earliness.

In the family of scheduling problems, tardiness and earliness are common parameters that have been widely defined and used [6], [7], [8].

Tardiness is the time elapsed between the deadline and the finishing time of a task, which represents the delay in task execution. The earliness of a task is defined as:

$$E_i = \max(0, f_i - h_i) \quad (1)$$

We denote the prescribed due date for the task by h_i , in other words, f_i is the deadline for completion of the task.

With the above notation, tardiness is defined as:

$$T_i = \max(0, h_i - f_i) \quad (2)$$

Let's look at a specific example of the harmful effects of completing a project early in the frozen food industry. If the product is made too early, the manufacturer must bear additional costs for renting a warehouse for storage. In addition, when products reach users, their shelf life is reduced.

In the past, there have been studies on tardiness and earliness applied to scheduling problems, but these two factors have never been applied to a multi-skill scheduling problem like MS-RCPSP. The new contribution of this

paper compared to previous studies is to consider delay and earliness as the objective function for the MS-RCPSP problem, instead of makespan or cost like previous studies [3], [4],[5].

This paper proposes a novel problem called E-RCPSP (Earliness tardiness MS-RCPSP), which is the problem of finding a project schedule that minimizes total tardiness and earliness. Figure 1 depicts a project schedule consisting of 5 tasks, whereby the total delay and advance of this schedule is $6 + 8 + 5 + 15 + 6 = 40$.

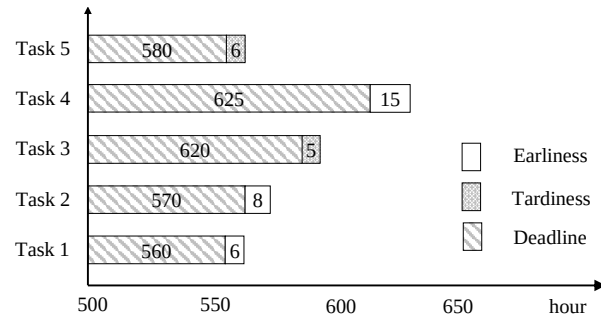


Figure 1. Tardiness and earliness of a project consisting of 5 tasks.

The rest of the paper is structured as follows. Section 2 introduces previous research related to the MS-RCPSP problem as well as the concepts of delay and earliness. Section 3 presents the mathematical model of the E-RCPSP problem. Section 4 introduces two solutions to solve the proposed problem including the GA-E algorithm and DE-E differential evolution algorithm. To verify the effectiveness of the two proposed algorithms, section 5 presents experimental results, and analyzes and evaluates them to clarify the performance of those algorithms.

II. RELATED WORKS

1. Approximate Algorithm for MS-RCPSP Problem

Myszkowski solved the original version of the MS-RCPSP problem using a variety of methods. Initially, he used a heuristic method of sorting tasks based on the order of their processing time, and then assigning resources in that order. Later, Myszkowski turned to meta-heuristic algorithms such as the Ant Colony algorithm [3] and the GA algorithm [5] and obtained higher-quality solutions. In addition to proposing algorithms, another important contribution of Myszkowski et al. is the iMOPSE dataset [2]. This dataset is recognized as a standard benchmark and is therefore widely used by many other studies to verify the performance of algorithms for the MS-RCPSP problem.

A variant of MS-RCPSP was named MMSRCPS (Multi-mode Multi-skilled Resource-Constrained Project

Scheduling Problem) by Hosseinian [9]. This variant of the MS-RCPSP problem adds the constraint that each task can only be executed in certain modes, and the mode cannot be changed once the task's execution has been started.

To find an efficient schedule for the MMSRCPSP problem, Hosseinian et al. [10] used a traditional GA algorithm combined with a decision-making method based on the Shannon-entropy information measure to search for better schedules for the next generation. Then, with the original MS-RCPSP problem, Hosseinian turned to using the Dandelion Algorithm [11] and verified its effectiveness on the iMOPSE dataset. In this research, Hosseinian solved the multi-objective MS-RCPSP problem with two objectives: makespan and execution cost.

Similar to Hosseinian, H. Davari-Ardakani [1] also studied a multi-objective variant of the MS-RCPSP problem with an objective function including two components: project implementation time and cost. In the proposed problem, called MSPSP, H. Davari-Ardakani only considers projects with the following two characteristics:

- Project implementation time is arbitrary. The project can even be done in the evening or on weekends.
- Energy costs are enormous, equal to employee salaries.

To solve the MS-RCPSP problem in a Grid environment, Youni and colleagues proposed a hybrid heuristic algorithm based on a task arrangement mechanism [12]. The works of Maghsoudlou [13] and Bibiks [14] are similar in that they apply the Cuckoo Search algorithm to search for the optimal schedule of multi-risk projects based on three different evaluation objectives. Zhu et al. proposed an evolutionary algorithm based on the multi-verse algorithm [15].

2. Solutions to Minimize Earliness and Tardiness

Mario Vanhoucke and colleagues [16] proposed the RCPSPWET problem, an extension of the RCPSP problem. The goal of the RCPSPWET problem is to minimize the total Earliness-Tardiness penalty cost. Mario Vanhoucke uses prioritization constraints between tasks and constraints on continuously available renewable resources. The authors solved the RCPSPWET problem using the search tree algorithm and branching algorithm.

Yazdan Khoshjahan [6] proposed a new problem, named RCPSP-DET, whose goal is to minimize the net present value of project Earliness-Tardiness penalties. The author solves this problem through a meta-heuristic algorithm.

Maarten Otten [17] minimized the Earliness-Tardiness cost across multiple machines for surgical scheduling. The author has proposed a solution to minimize two factors: temporary time and overtime of doctors and medical staff.

Saheed Akande [18] proposed three versions of the RCPSP problem related to earliness and tardiness. The first version of the problem deals with the total costs incurred due to tardiness work. The second version of the problem maximizes earliness and minimizes tardiness during project execution. The third version of the problem minimizes the linear objective function of earliness and tardiness. The results obtained from the three versions of the problem represent a measure of the degree of penalty arising from each type of problem. In another paper, Saheed Akande solved the multi-objective scheduling problem, including minimum total completion time, maximum tardiness, and earliness [19].

The general limitation of the above studies is that they only aim to reduce the total early-late penalty cost of the RCPSP problem; There is no solution for minimizing the early-late penalty on the MS-RCPSP problem.

III. PROBLEM FORMULATIONS

Given a set of tasks (denoted by Job) to be executed that are related to each other in order of priority, meaning that the child task is only allowed to start after the parent task has been completed. Given a set of resources (denoted by Res), each resource possesses different skills. To execute task Job_i , resource Res_i must have the skills that match Job_i 's requirements and one skill level greater than or equal to the skill level required by Job_i .

The goal of the E-RCPSP problem is to find a project schedule that minimizes total earliness and tardiness. Earliness and tardiness are defined by formulas (1) and (2), respectively. E-RCPSP can be conceptually formulated based on the following notations:

- Tas_i : set of tasks that need to be completed before the time task i is executed.
- Fis : the set of skills.
- Fis^i : the subset of skills owned by resource i , $Fis^i \subseteq Fis$.
- Fis_i : skill i .
- tim_j : the time required to perform task j .
- Res : the set of resources.
- Res^m : the set of resources that could handle task m , $Res^m \subseteq Res$.
- Res_i : resource i .
- Job : the set of all resources.
- Job^k : the set of tasks that resource k could handle, $Job^k \subseteq Job$.
- Job_i : task i .
- req^i : set of skills required to perform the task i .
- Sta_k, Clo_k : start time and finish time of task k .

- $Var_{u,v}^t$: a boolean variable; it equals 1 means that task u will be executed by resource v at time t , it equals 0 in other cases.
- lev_i : the level of skill i .
- gen_i : type of skill i .
- mak : makespan (execution time) of schedule.
- TSc : a candidate schedule.
- TSc_{all} : the set of candidate schedules.
- $f(TSc)$: makespan (execution time) of scheduled TSc .
- cou : number of tasks, num : number of resources.

E-RCPSP problem could be defined as follows:

$$f(TSc) \rightarrow \min$$

where

$$f(TSc) = \sum_{i=1}^n (e_i E_i + t_i T_i) \quad (3)$$

e_i and t_i are the penalty units for the earliness and tardiness times of task i , respectively. E_i , T_i are the earliness and the tardiness of task i , respectively.

Subject to:

$$Fis^k \neq \emptyset \quad \forall Res_k \in Res \quad (4)$$

$$tim_i \geq 0 \quad \forall Job_i \in Job \quad (5)$$

$$Clo_i \geq 0 \quad \forall Job_i \in Job \quad (6)$$

$$Clo_i \leq Clo_k - tim_k \quad \forall Job_k \in Job, k \neq 1, Job_i \in Tas_k \quad (7)$$

$$\forall Job_i \in Job^k, \exists Fis_q \in Fis^k : gen_{Fis_q} = gen_{r_i}, \quad lev_{F_q} \geq lev_{r_i} \quad (8)$$

$$\forall Res_k \in Res, \forall q \in mak : \sum_{i=1}^n Var_{i,k}^q \leq 1 \quad (9)$$

$$\forall Job_i \in Job, \exists !q \in [0, mak], \quad (10)$$

$$\exists !Res_k \in Res : Var_{i,k}^q = 1 \text{ with } Var_{i,k}^q \in \{0, 1\}$$

Note that:

- Formulation (4) means that every resource has at least one skill.
- Formulation (5, 6) means that task execution time is greater than or equal to 0.
- Formulation (7) forces the parent task to be finished before the start time of the children's task.
- Formulation (8) means that for every task, there is always at least one resource that has enough skill level to handle that task.
- Formulation (9) ensures that each resource (k) can only perform at most one task at any time (q).
- Formulation (10) means that each task is assigned to a single resource and executed only by that resource.

IV. PROPOSED ALGORITHM

1. DE algorithm for E-RCPSP problem

Differential Evolution (DE) is the information-driven evolutionary algorithm introduced by R. Storn and K. Price [20]. DE is the directed evolutionary algorithm that uses mutation methods to find better solutions in the next generation. DE has been applied effectively to solve NP-Hard problems.

This paper proposes a DE-E which is algorithm to solve the E-RCPSP problem, its operating steps are presented in Figure 2 and the source codes are detailed in Algorithm 1 below.

Algorithm 1: DE-E algorithm

Data: N (size of population), numGen (number of generations)
Result: the best schedule

```

1 Load and valid dataset; count = 0; P = initialize(N);
2 gbest = fbest(P); /*the best individual in the population*/
3 while count < numGen do
4   for i < N do
5      $x_{r_1} \neq x_{r_2} \neq x_{r_3} \neq x_i$ ; /*Random different individuals*/
6      $F \leftarrow \text{rand}(0, 1)$ ; /*mutation constant*/
7      $v_i(t) = x_{r_1} + F * (x_{r_2} - x_{r_3})$ ; /*mutation*/
8      $I_{rand} = \text{randi}(1, tCount)$ ; /*Irand ensures  $x_i \neq v_i$ */
9      $CR \in [0, 1]$ ; /*probability of crossover*/
10    for j < tCount do
11      if  $i = I_{rand}$  OR  $\text{rand}(0, 1) \leq CR$  then
12         $u_{i,j} = v_{i,j}$ ;
13      end
14      else
15         $u_{i,j} = x_{i,j}$ ;
16      end
17    end
18     $u1 = \text{simplebound}(u)$ ;  $z = f(u1)$ ;  $fxi = f(x_i)$ ;
19    /*select population*/
20    if  $fxi < z$  then
21       $P[i] = u1$ ;
22    end
23    /*update makepan*/
24    if  $z < \text{makespan}$  then
25       $\text{makespan} = z$ ;
26      CopyMPS( $u1$ );
27    end
28    if  $z < \text{fitness}[i]$  then
29       $\text{fitness}[i] = z$ ; /*update fitness*/
30    end
31  end
32  /*update gbest*/
33  gbest = fbest(P); count = count + 1;
34 end
35 return gbest;
```

Where, some functions are used in Algorithm 1:

f : objective function of E-RCPSP problem.

$fbest$: the function returns the best individual in the population. $tCount$: number of tasks.

$simplebound$: validate current particle/individual.

$CopyMPS$: update population.

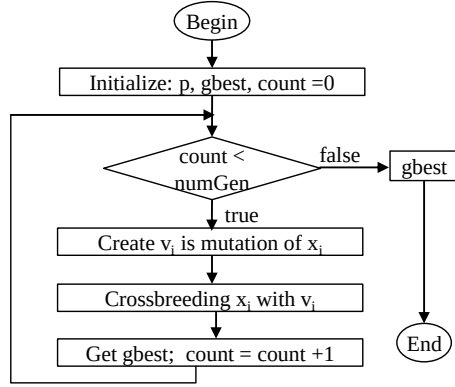


Figure 2. DE-E algorithm for the E-RCPSP problem.

2. GA algorithm for E-RCPSP problem

Genetic algorithm (GA) is a computer science technique that seeks to find the best solution to combination optimization problems. GA is an evolutionary algorithm that applies principles of evolution such as mutation, natural selection, and crossover. This paper proposes a GA-E, which is a GA algorithm for E-RCPSP problem, its operating steps are presented in Figure 3. GA-E algorithm are details as follows.

Algorithm 2: GA-E algorithm

Data: N (size of population), numGen (number of generations)
Result: the best schedule

```

1 Load and valid dataset;
2 count = 0;    p = initialize(N);    gbest = fbest(p);
3 while count < numGen do
4     initialize(fitness);
5     /*Evaluate fitness*/
6     for i < N do
7         fitness[i] = f(p[i]);
8         if fitness[i] < makespan then
9             makespan = fitness[i];    CopyMPS(p[i]);
10        end
11    end
12    /*select population*/
13    for i < N do
14        ichon1 = rand(1,N);    ichon2 = rand(1,N);
15        if fitness[ichon1] > fitness[ichon2] then
16            Pselect [i] = p[ichon1];
17        end
18        else
19            Pselect [i] = p[ichon2];
20        end
21    end
22    /* Crossover */
23    Crossbreeding of randomly selected individuals;
24    /*mutation*/
25    muta = 0.1*rand(0,1);    /*mutation probability*/
26    Create mutation with mutation probability "muta"
    and get Pselect;
27    p = Pselect;    count = count+1;
28 end
29 return fbest(p);
    
```

Some functions are used in algorithm 2:

f : objective function of E-RCPSP problem.

$fbest$: the function returns the best individual in the population.

$CopyMPS$: update population.

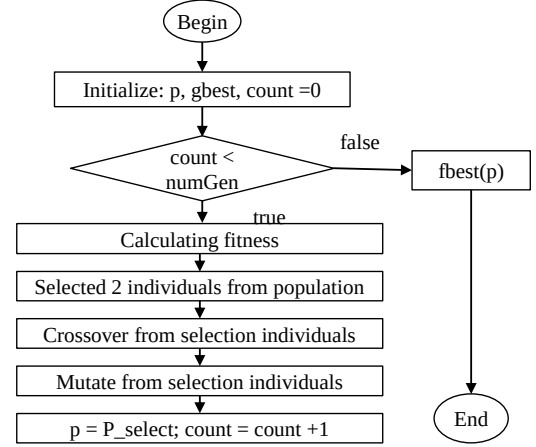


Figure 3. Proposed GA-E algorithm for E-RCPSP problem

V. PERFORMANCE EVALUATION

1. Experimental settings

To verify the performance of the proposed algorithms, experiments are conducted on C# Visual Studio 2022, processor Core i5 2.4GHz, RAM 6GB, OS Windows 11.

Experiments were conducted iMOPSE dataset [2]. As detailed in Table I, iMOPSE's data is stored in the file .def, which includes the following information fields:

- Number of the project's tasks and resources.
- Priority relationship between tasks.
- The number of relationships between tasks.
- Number of skills owned by resources.
- The subset of skills owned by each resource.

2. Experimental Results

To find the optimal schedule for the E-RCPSP problem, this article proposes two algorithms DE-E and GA-E. Experiments were conducted to evaluate the performance of the proposed algorithms. The experimental results of the two algorithms are described in Table II and Table III, respectively, in which column E represents the earliness value, column T represents the tardiness value and column ET represents the value of the objective function, all of these values are in seconds.

Figure 4 shows the experimental results of the GA-E algorithms with the number of generations being 1000,

Table I
iMOPSE DATASET

No.	Dataset	Name	Tasks	Resources	Precedence Relations	Skills
1	iM1	100_5_22_15	100	5	22	15
2	iM2	100_5_46_15	100	5	46	15
3	iM3	100_5_48_9	100	5	48	9
4	iM4	100_5_64_15	100	5	64	15
5	iM5	100_5_64_9	100	5	64	9

Table II
EXPERIMENTAL RESULTS OF THE GA-E ALGORITHM

Dataset	1000 generations			3000 generations			5000 generations		
	E(s)	T(s)	ET(s)	E(s)	T(s)	ET(s)	E(s)	T(s)	ET(s)
iM1	4148	3607	7755	5992	3210	9202	6501	2203	8704
iM2	3547	1936	5483	4850	1017	5867	3113	2359	5472
iM3	2543	2332	4875	2219	2731	4950	3711	1236	4947
iM4	2889	1836	4725	1197	2892	4089	2642	1304	3946
iM5	1052	1950	3002	1363	1479	2842	1102	1408	2510

Table III
EXPERIMENTAL RESULTS OF THE DE-E ALGORITHM

Dataset	1000 generations			3000 generations			5000 generations		
	E(s)	T(s)	ET(s)	E(s)	T(s)	ET(s)	E(s)	T(s)	ET(s)
iM1	459	513	972	530	114	644	908	797	1705
iM2	860	1058	1918	1965	378	2343	539	541	1080
iM3	1548	1030	2578	1358	1153	2511	1151	757	1908
iM4	1594	1497	3091	557	504	1061	575	1221	1796
iM5	2663	1808	4471	2665	1777	4442	2681	1732	4413

3000, and 5000. Figure 5 shows the experimental results of DE-E with the corresponding number of generations. Through this comparison, it can be seen that the GA-E algorithm is more effective when performed on datasets that have a high number of precedence relations and a small number of skill levels. In other words, for the same number of tasks and resources, on a dataset with a higher number of precedence relations and a lower number of skill levels, the GA-E algorithm finds better results than DE-E.

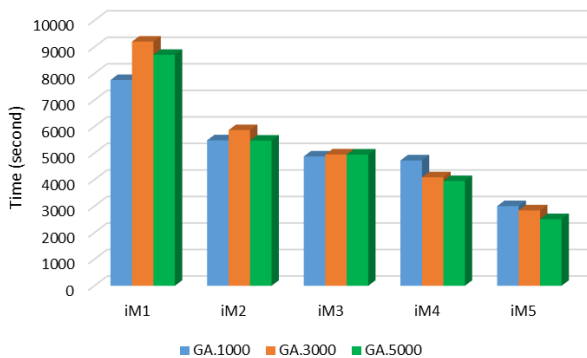


Figure 4. The efficiency of the GA-E algorithm

Figure 5 depicts that the DE-E algorithm is more effective when the dataset has a small number of precedence relations and a large number of skill levels. That is, with

the same number of tasks and resources, on a dataset with a smaller number of precedence relations and a larger number of skill levels, the results of the DE-E algorithm are better than the GA-E.

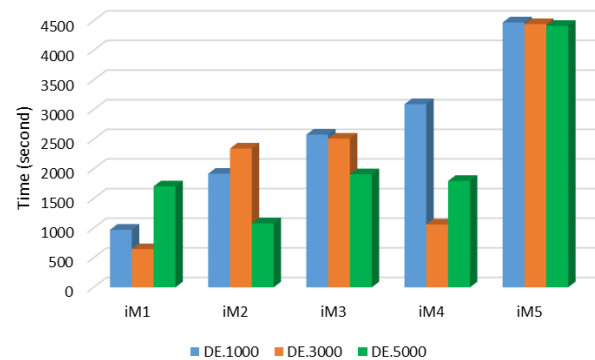


Figure 5. The efficiency of the DE algorithm

Experimental results with different numbers of generations (1000, 3000, 5000) show that the results of the DE-E algorithm are better than GA-E when performed on the iM1, iM2, iM3, and iM4 datasets. In contrast, the GA-E algorithm outperforms DE-E on the remaining datasets. A comparison of experimental results of GA-E and DE-E algorithms with 1000 generations is depicted in Figure 6:

- With the iM1 dataset, the ET value of DE-E is only 12.53% of the GA-E.
- With the iM2, iM3, iM4, iM5 dataset, the ET value of DE-E is 34.98%, 52.88%, 65.42%, and 148.93% of GA-E, respectively.

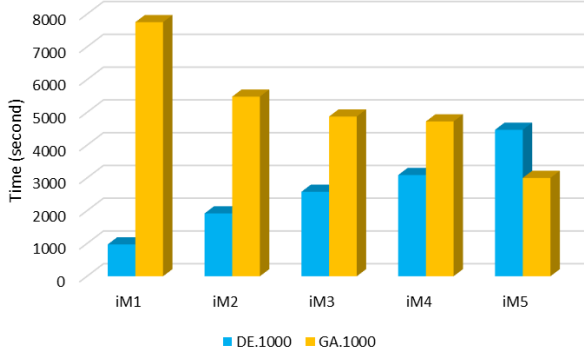


Figure 6. The effectiveness of GA-E and DE-E when experimenting with 1000 generations

A comparison of experimental results of GA-E and DE-E algorithms with 3000 and 5000 generations is depicted in Figure 7 and Figure 8, respectively.

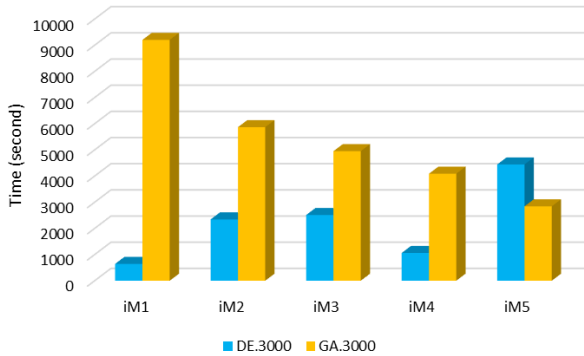


Figure 7. The effectiveness of GA-E and DE-E when experimenting with 3000 generations

A comparison of experimental results of GA-E and DE-E algorithms with 5000 generations is depicted in Figure 8:

- With the iM1 dataset, the ET value of DE-E is only 19.59% of the GA-E.
 - With the iM2, iM3, iM4, iM5 dataset, the ET value of DE-E is 39.94%, 50.73%, 25.95%, and 156.30% of GA-E, respectively.
- A comparison of experimental results of GA-E and DE-E algorithms with 5000 generations is depicted in Figure 8:
- With the iM1 dataset, the ET value of DE-E is only 19.59% of the GA-E.

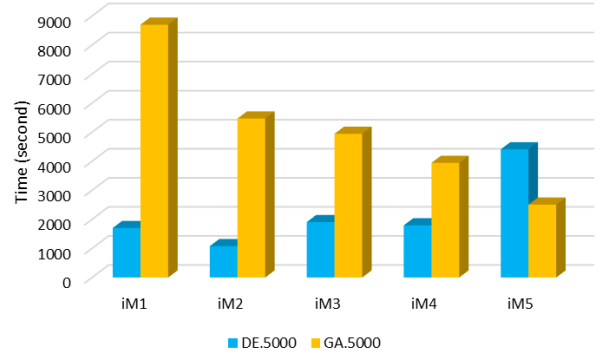


Figure 8. The effectiveness of GA-E and DE-E when experimenting with 5000 generations

- With the iM2, iM3, iM4, iM5 dataset, the ET value of DE-E is 19.74%, 38.57%, 45.51%, and 175.82% of GA-E, respectively.

VI. CONCLUSION AND FUTURE WORK

Earliness and tardiness are very important factors that allow evaluation of the effectiveness of schedules. In the family of scheduling problems, the MS-RCPSP problem has many practical applications, so it attracts the attention of many researchers. However, the application of earliness and tardiness to the MS-RCPSP problem has never been mentioned in previous studies.

This research has built the foundation for project scheduling so that the execution of tasks matches or does not deviate much from the scheduled deadline. We proposed a new problem called E-RCPSP and officially stated its mathematical model. Two evolutionary algorithms based on GA and DE have been built to find the optimal schedule for the proposed problem. Experimental results show that among the two proposed algorithms, each has its own advantages, meaning that no algorithm outperforms the other on all datasets.

In the future, the authors hope to find a comprehensive solution capable of finding the optimal schedule in all cases. To achieve that goal, we plan to apply deep reinforcement learning (DRL) to help the proposed algorithm learn from the knowledge of previous generations to make correct decisions for the next generations.

ACKNOWLEDGMENT

This research was supported by the Ministry of Education and Training, Viet Nam, under the science and technology project no. B2024-SPH-14 supervised by the Hanoi National University of Education (HNUE), Viet Nam.

REFERENCES

- [1] H. Najafzad, H. Davari-Ardakani, R. Nemati-Lafmejani, "Multi-skill project scheduling problem under time-of-use electricity tariffs and shift differential payments", *In Energy Journal*, vol. 168, pp. 619-636, Elsevier, 2019.
- [2] P.B. Myszkowski, M. Laszczyk, I. Nikulin, M. Skowro, "iMOPSE: a library for bicriteria optimization in Multi-Skill Resource-Constrained Project Scheduling Problem", *In Soft Computing Journal*, 23: 32397, 2019.
- [3] P.B. Myszkowski, M. Skowroński, L. Olech, K. Oślizło, "Hybrid Ant Colony Optimization in solving Multi-Skill Resource-Constrained Project Scheduling Problem", *In Soft Computing Journal*, Volume 19, Issue 12, pp.3599-3619, 2015.
- [4] S. Javanmard, B. Afshar-Nadjafi, S.T. Niaki, "Preemptive multi-skilled resource investment project scheduling problem: Mathematical modeling and solution approaches," *In Computers & Chemical Engineering*, 96, pp. 55-68, 2017.
- [5] Skowronski M., Myszkowski P.B., "Specialized genetic operators for Multi-Skill Resource-Constrained Project Scheduling Problem", *In 19th Inter. Conference on Soft Computing - Mendel 2013*, pp. 57-62, 2013
- [6] Y. Khoshjahan, A.A. Najafi, B.A. Nadjafi, "Resource constrained project scheduling problem with discounted earliness-tardiness penalties: Mathematical modeling and solving procedure", *In Computers & Industrial Engineering*, 2013.
- [7] V. A. Strusevich, K. Rustogi, "Scheduling with Times-Changing Effects and Rate-Modifying Activities", *In Springer International Publishing*, 2017.
- [8] Kramer, A., Subramanian, A., "A unified heuristic and an annotated bibliography for a large class of earliness-tardiness scheduling problems", *In Journal of Scheduling*, vol. 22, 2019.
- [9] A.H. Hosseini, V. Baradaran, "An Evolutionary Algorithm Based on a Hybrid Multi-Attribute Decision Making Method for the Multi-Mode Multi-Skilled Resource-Constrained Project Scheduling Problem", *In Journal of Optimization in Industrial Engineering*, 12.2, pp. 155-178, 2019.
- [10] A.H. Hosseini, V. Baradaran, "Detecting communities of workforces for the multi-skill resource-constrained project scheduling problem: A dandelion solution approach", *In Journal of Industrial and Systems Engineering*, pp. 72-99, 12.2019.
- [11] A.H. Hosseini, V. Baradaran, "P-GWO and MOFA: two new algorithms for the MSRCPS with the deterioration effect and financial constraints (case study of a gas treating company)", *In Applied Intelligence*, 50, pp. 2151-2176, 2020.
- [12] Younis, Muhanad Tahrir, and Shengxiang Yang, "Hybrid meta-heuristic algorithms for independent job scheduling in grid computing", *In Applied soft computing*, vol.72, pp. 498-517, 2018.
- [13] Maghsoudlou, Hamidreza, Behrouz Afshar-Nadjafi, and Seyed Taghi Akhavan Niaki, "Multi-skilled project scheduling with level-dependent rework risk; three multi-objective mechanisms based on cuckoo search", *In Applied Soft Computing*, vol. 54, pp. 46-61, 2017.
- [14] Bibiks, Kirils, et al., "Improved discrete cuckoo search for the resource-constrained project scheduling problem", *In Applied Soft Computing*, vol. 69, pp. 493-503, 2018.
- [15] Zhu, Lei, Jian Lin, Zhou-Jing Wang, "A discrete oppositional multi-verse optimization algorithm for multi-skill resource constrained project scheduling problem", *In Applied Soft Computing*, vol. 85, 2019.
- [16] Mario Vanhoucke, Erik Demeulemeester, and Willy Herroelen, "An Exact Procedure for the Resource-Constrained Weighted Earliness-Tardiness Project Scheduling Problem", *In Kluwer Academic Publishers. Manufactured in The Netherlands*, 2001.
- [17] Maarten Otten, Aleida Braaksma, Richard J. Boucherie, "Minimizing Earliness/Tardiness costs on multiple machines with an application to surgery scheduling", *Operations Research for Health Care*, vol. 22, 2019.
- [18] S. Akande, G. Ajisegiri, "Three classes of Earliness-Tardiness (E/T) scheduling problems", *In Fuel Communications*, vol. 10, 2022.
- [19] S. Akande, Ajisegiri GO, "Single machine multi-criteria scheduling problem: total completion time, maximum lateness, and maximum earliness performance measures", *In International Journal of Planning and Scheduling*, vol. 3, 2021.
- [20] K.Price, R.Storn, J.Lampinen: "Differential Evolution - A Practical Approach to Global Optimization", *In Springer, Berlin, Germany*, 2005.



Hao Nguyen Thi received Bachelor degree at Hung Vuong University, Viet Nam, in 2009. She received M.S. degree at Ha Noi University of Science and Technology, Viet Nam, in 2013. She worked at Hung Vuong University, Phu Tho, Viet Nam from 2009. Her research interests include optimization algorithms, approximation algorithms, deep

learning.

Email: haont@hvu.edu.vn



Huu Dang Quoc received Bachelor and M.S. degree in School of Information Technology, Vietnam National University, Ha Noi, Viet Nam, in 2000 and 2015. He received Ph.D. degree in Military Institute of Science and Technology, Ha Noi, Vietnam, Viet Nam, 2022. He worked at Thuong Mai University, Ha Noi, Viet Nam from 2006.

His research interests include Computer Network and Software Engineering, Evolution Algorithm, Optimization Algorithm.

Email: huudq@tmu.edu.vn



Loc Nguyen The received Bachelor and M.S. degree in School of Information and Communication Technology, Hanoi University of Science and Technology, Viet Nam, in 1998 and 2001, respectively. He received Ph.D. degree in School of Information Science, Japan Advanced Institute of Science and Technology, Japan, 2007. He worked

at Hanoi National University of Education (HNUE), Ha Noi, Viet Nam from 1997 and is currently a professor, head of Department of Computer Engineering, Faculty of Information Technology, HNUE. His research interests include Optimization Algorithms, Approximation Algorithms, IoT, Networking Infrastructure.

Email: locnt@hnue.edu.vn

Phi-3 Meets Law: Fine-tuning Mini Language Models for Legal Document Understanding

Nguyen Huu Khanh¹, Nguyen Van Viet², Nguyen The Vinh², Nguyen Huu Cong¹

¹ Thai Nguyen University, Thai Nguyen, Vietnam ² Thai Nguyen University of Information and Communication Technology, Thai Nguyen, Vietnam

Correspondence: Khanh H. Nguyen, khanhnh@tnu.edu.vn

Received: 08/07/2024, revised: 26/08/2024, accepted: 16/09/2024

Digital Object Identifier: 10.32913/mic-ict-research-vn.v2024.n3.1281

Abstract: This study explored the application of Microsoft's Phi-3 model to legal document understanding. We fine-tuned Phi-3-mini-4k-instruct, a compact yet powerful language model, on CaseHOLD's 53,000+ multiple-choice questions designed to test legal case holding identification. Employing Low-Rank Adaptation (LoRA) and Quantized Low-Rank Adaptation (QLoRA) techniques, we optimized the fine-tuning process for efficiency. Our results demonstrated that the fine-tuned Phi-3-mini-4k-instruct model achieved an F1 score of 76.89, surpassing previous state-of-the-art models in legal document understanding. This performance was achieved with only 6000 training steps, highlighting Phi-3's rapid adaptability to domain specific tasks. The base Phi-3 model, without fine-tuning, achieved an F1 score of 64.89, outperforming some specialized legal models. Our findings underscored the potential of compact language models like Phi-3 in specialized domains when properly fine-tuned, offering a balance between model size, training efficiency, and performance. This research contributes to the growing body of work on adapting large language models to specific domains, particularly in the legal field.

Keywords: *Phi-3, CaseHold, fine-tuning, QLoRA, supervised fine-tuning*

Tiêu đề: Phi-3 với Luật pháp: Tinh chỉnh mô hình ngôn ngữ nhỏ trong việc hiểu văn bản pháp luật

Tóm tắt: Nghiên cứu này khám phá ứng dụng mô hình Phi-3 của Microsoft vào việc hiểu văn bản pháp lý. Chúng tôi đã tinh chỉnh Phi-3-mini-4k-instruct, một mô hình ngôn ngữ nhỏ gọn nhưng mạnh mẽ, trên hơn 53.000 câu hỏi trắc nghiệm của CaseHOLD được thiết kế để kiểm tra khả năng nhận dạng vụ án pháp lý. Sử dụng các kỹ thuật thích ứng bậc thấp (LoRA) và thích ứng bậc thấp lượng tử (QLoRA), chúng tôi đã tối ưu hóa quy trình tinh chỉnh để đạt hiệu quả. Kết quả của chúng tôi cho thấy mô hình Phi-3-mini-4k-instruct tinh chỉnh đạt điểm F1 là 76,89, vượt qua các mô hình tiên tiến trước đây về khả năng hiểu văn bản pháp lý. Hiệu suất này đạt được chỉ với 6000 bước đào tạo, làm nổi bật khả năng thích ứng nhanh chóng của Phi-3 với các tác vụ cụ thể theo miền. Mô hình Phi-3-mini-4k-instruct cơ bản, không cần tinh chỉnh, đã đạt điểm F1 là 64,89, vượt trội hơn một số mô hình pháp lý chuyên biệt. Kết quả thực nghiệm của chúng tôi nhấn mạnh tiềm năng của các mô hình ngôn ngữ nhỏ gọn như Phi-3 trong các miền chuyên biệt khi được tinh chỉnh đúng cách, mang lại sự cân bằng giữa quy mô mô hình, hiệu quả đào tạo và hiệu suất. Nghiên cứu này góp phần vào khối lượng công việc ngày càng tăng về việc điều chỉnh các mô hình ngôn ngữ lớn cho các miền cụ thể, đặc biệt là trong lĩnh vực pháp lý.

Từ khóa: *Phi-3, CaseHold, fine-tuning, QLoRA, supervised fine-tuning*

I. INTRODUCTION

Understanding legal documents is important in today's judicial and regulatory climate [1]. A legal document is any text that has legal significance or is used in a legal context to communicate rights, responsibilities, or procedures [2]. Legal practitioners, politicians, and researchers increasingly rely on digital texts, making it critical to comprehend legal papers swiftly and accurately. However, the language

used in these documents is often characterized by specialized vocabulary, complex sentence structures and formal. Statutes, case law, contracts, and regulatory filings are well-known examples for their complex structure, specialized terminology, and sophisticated logic. Moreover, accurately interpreting legal language requires not only linguistic proficiency but also an understanding of legal principles, rules of interpretation and precedent. Thus, understanding legal documents is a challenging task for both human and

machines to correctly comprehend [3–5].

Numerous studies were conducted to address the challenges of understanding legal documents using various approaches. For example, pioneer techniques relied on rule-based systems [6–8], which were accurate but time-consuming and difficult to scale. More recent advances used machine learning techniques, namely as pre-trained language models, to automate and improve the comprehension of legal documents. These pre-trained models, including K-BERT [9], Lawformer [10], JuriBert [11], jurBert [12], Paraformer [13], XLM-RoBERTa [14] and their derivatives, although exhibited promising performance in the legal domain, they all shared the same challenge as lack of understanding of natural language due to being trained on small datasets. This problem could be alleviated by utilizing large language models (e.g., Llama, GPT). However, these oversized models are primarily accessible to big tech companies (e.g., Google and Microsoft) where computational resources are available, making it challenging for researchers to conduct experiments [15]. This issue is not limited to the legal domain but extends to other similar research settings [16]. Therefore, there is a strong need for a mid-to-mini-sized pre-trained model that can effectively balance natural language understanding and model size [16, 17], particularly when specialized in the legal documents domain.

In response to this need, Microsoft recently released Phi-3 [18] on March 22, 2024, as pre-trained mid-to-mini-sized models. However, to the best of our knowledge, there is no evidence demonstrating the performance of Phi-3 in the legal document domain. Therefore, the purpose of this research is to address this gap by conducting an experiment to evaluate the efficacy of Phi-3 on legal documents.

To meet that objective, we fine-tuned Phi-3-mini-4k with the CaseHOLD [19] dataset, a benchmark intended exclusively for assessing legal document understanding. Figure 1 presents overview of process in this study. Our goal is to experiment Phi-3-mini-4k-instruct and evaluate its potential for comprehending and processing legal texts. This study intends to add to the body of knowledge in legal Natural Language Processing (NLP) by giving empirical evidence on the performance of a mini size pre-trained model, thus opening the way for more efficient and accessible legal document analysis applications.

II. MATERIALS AND METHODS

1. The base model

In our study, we used Phi-3 (introduced by Microsoft on March 22, 2024) as the base model for the fine-tuning task [18]. Compared to many state-of-the-art models, Phi-3 is notable for its high-quality parameters and size. The

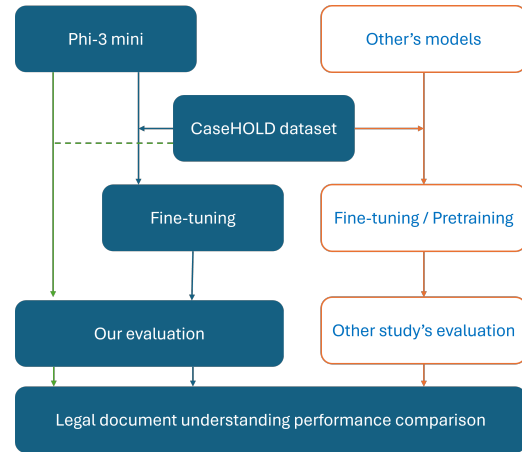


Figure 1. Overview of process in this study

term "high-quality parameters" refers to high-quality data curation, extensive post-training, and reinforcement learning from human feedback (RLHF). Although the number of parameters is relatively small compared to other large language models, evidence [18] shows that Phi-3's performance surpasses that of larger models (e.g., Mixtral 8x7B and GPT-3.5). There is currently no standardized definition for terms like "mini" or "small" models in the literature. However, in the field of AI or as in our study, models can be classified based on the number of parameters: "tiny" refers to models with fewer than 1 billion parameters, "mini" ranges from 1 to 5 billion parameters, and "small" covers models with 5 to 10 billion parameters, following the terminology used by Microsoft. Phi-3 was released in several versions including Mini (3.8B parameters), Small (7B parameters), Medium (14B parameters) and Vision (4.2B parameters). In this study, we chose Phi-3-mini-4k-instruct model from the Phi-3 family due to several compelling factors. First, despite its compact size, Phi-3-mini-4k-instruct has demonstrated impressive performance comparable to significantly larger models, suggesting strong language understanding capabilities. This balance of size and performance is particularly attractive for resource-constrained settings. Second, Phi-3's training methodology aligns well with the requirements of legal document analysis, where accurate and nuanced language understanding is paramount. We hypothesize that these characteristics would translate well to the legal domain, making Phi-3-mini-4k-instruct a strong candidate for fine-tuning on the CaseHOLD dataset.

2. CaseHold Dataset

The CaseHOLD dataset serves as a benchmark for evaluating legal document understanding in the U.S. legal domain. It comprises over 53,000 multiple-choice questions

Table I
EXAMPLE OF CASEHOLD DATA ITEM

Citing_prompt	CURIAM. Affirmed. See Adams v. State, 76 So.3d 367 (Fla. 3d DCA 2011) (<HOLDING>); accord Little v. State, 77 So.3d 722 (Fla. 3d
Holding_0	holding that section 89313 as amended by section 893101 florida statutes 2002 is constitutional
Holding_1	holding that the legislature amended section 8120142d florida statutes now renumbered as section 8120143c in 1992 to omit habitual offender penalties for the crime of felony petit theft
Holding_2	recognizing that the 1992 amendment to section 8930216 florida statutes changed the law
Holding_3	holding florida sexual predators act section 77521 florida statutes 2000 to be unconstitutional as violating procedural due process
Holding_4	holding that arbitration is not a civil action as that term is used in section 76873 florida statutes 1991
Label	0

designed to identify the relevant holding of a cited case. Each question presents a citing context from a judicial decision as the prompt (named "Citing_prompt"), five answer choices derived from holding statements within legal citations (named "holding_[0-4]") and an indicator for correct answer (named "label"). Table I shows an example of CaseHold data item. The correct answer corresponds to the citing text, while the four incorrect options are selected based on their similarity, making the task challenging. Constructed from the Harvard Law Library case law corpus, this dataset offers a robust framework for fine-tuning and evaluating advanced language models. The precision in extracting holding statements and the strategic selection of distractors highlight its value in advancing legal AI research.

3. Experimental setup

a) Data preprocessing

The CaseHOLD dataset, comprising fields like Citing_prompt, five holdings, and a label, required careful preprocessing to align with Phi-3's fine-tuning instructions. Our first step involved creating a function to structure the data into a conversational format, integrating all fields effectively. This structured data was then passed to the Phi-3 tokenizer's apply_chat_template function, adhering to the "<user>Question<end><assistant>Answer<end>" format for training, example of chat format is shown in Table II. During evaluation, we adjusted the process by sending only the Citing_prompt and the holdings, omitting the label to allow the model to predict the correct answer. The instruction for apply_chat_template in this phase was simplified to "<user>Question<end><assistant>", facil-

Table II
EXAMPLE CHAT FORMAT FOR TRAINING

<pre><s><user> Choose the best option to fill <HOLDING> based on this context, answer only with the number of option(for example: 0): CURIAM. Affirmed. See Adams v. State, 76 So.3d 367 (Fla. 3d DCA 2011) (<HOLDING>); ac-cord Little v. State, 77 So.3d 722 (Fla. 3d Options to choose: 0. holding that section 89313 as amended by section 893101 florida statutes 2002 is constitutional 1. holding that the legislature amended section 8120142d florida statutes now renumbered as section 8120143c in 1992 to omit habitual offender penalties for the crime of felony petit theft 2. recognizing that the 1992 amendment to section 8930216 florida statutes changed the law 3. holding florida sexual predators act section 77521 florida statutes 2000 to be unconstitutional as violating procedural due process 4. holding that arbitration is not a civil action as that term is used in section 76873 florida statutes 1991 <endl><lassistant> 0 <endl></pre>
--

itating an accurate assessment of the model's predictive capabilities in a realistic legal question-answering context.

b) LoRA

There are various fine-tuning approaches in the literature, ranging from full fine-tuning to partial fine-tuning [20]. Among these, Low-Rank Adaptation (LoRA) [20] is an innovative fine-tuning technique widely adopted by the AI community. Compared to other techniques that are resource-intensive and introduce latency, LoRA addresses these issues by training only the "intrinsic dimension" and then updating the original weights accordingly. Mathematically, LoRA could be expressed as follows [20]:

$$W = W_0 + \Delta W = W_0 + BA$$

$$B \in \mathbb{R}^{d \times r}, \quad A \in \mathbb{R}^{r \times k}$$

Where:

- W is the updated weight matrix.
- W_0 is the original pre-trained weight matrix.
- ΔW is the low-rank update applied to the original weights.
- B and A are low-rank matrices.
- r is the rank, a much smaller dimension than d or k .

The true value of LoRA lies in training matrix A and B instead of full training the original pretrained weight W_0 . As such, LoRA significantly reduces the computational resources (i.e., training on low-rank matrices) required for fine-tuning while maintaining performance (i.e., through W). This approach enables efficient adaptation allowing researchers to explore specialized applications such as legal document understanding without the need for extensive hardware resources. In our experiment, B and A are low-rank matrices with a rank (r) of 16, which means the updates only happen in a smaller 16-dimensional space. We applied an alpha value of 16, which scales the updates

to ensure they have a noticeable effect on the model's performance. To prevent the model from overfitting, we used dropout with a rate of 0.05 (5%), meaning a small portion of updates was randomly ignored during training to help the model generalize better. The LoRA updates were applied to key parts of the model, including the attention mechanism (key (k_proj), query (q_proj), value (v_proj), and output (o_proj) projections) and the feed-forward networks (gate ('gate_proj'), down (down_proj), and up (up_proj) projections). This setup balances adaptation capacity with parameter efficiency, allowing for effective fine-tuning of the Phi-3-mini-4k-instruct model on the CaseHOLD legal dataset.

c) QLoRA

To speed up the training process, we utilized Quantized Low-Rank Adaptation (QLoRA) technique [21]. The basic idea of QLoRA is to reduce the precision of the model's parameters (weights) by converting them from high precision (e.g., 32-bit floating point) to a lower precision format (e.g., 8-bit or even lower). As such, it reduces the memory required to store the model and speeds up computation. In our study, we set the model to implement 4-bit quantization (load_in_4bit=True) with the "nf4" quantization type, balancing compression and accuracy. We set the compute dtype to bfloat16 for improved training stability and employed double quantization (bnb_4bit_use_double_quant=True) to further optimize memory usage. This setup allowed us to leverage the full potential of Phi-3 while minimizing resource requirements.

d) Supervised fine-tuning

Supervised fine-tuning (SFT) was originally proposed by Ouyang et al. [22]. It is an important stage in aligning LLMs that involves training them on a supervised dataset of high-quality outputs. This procedure involves fine-tuning the model with a next token prediction aim, similar to pre-training, but with a focus on simulating desired behaviors. SFT is computationally efficient and improves the model's performance in terms of instruction compliance, coherence, and accuracy. Given our limited computational resources and the supervised nature of the CaseHOLD dataset, we determined that SFT is particularly well-suited for our study. Accordingly, we employed supervised fine-tuning using an easy-to-use API of Hugging Face, named Supervised Fine-tuning Trainer (SFTTrainer). This API facilitated the creation of our SFT model, optimizing the use of our available resources.

e) Training hyperparameters

In fine-tuning Phi-3-mini-4k-instruct on the CaseHOLD dataset, we carefully selected a set of hyper parameters to optimize the model's performance in legal document

understanding. We employed a relatively low learning rate of 0.0001 to ensure stable training, coupled with small batch sizes to accommodate the model's size and memory constraints. The Adam optimizer was used with standard beta values, while a linear learning rate scheduler with a 10% warm-up ratio helped manage the learning rate throughout training. We ran the fine-tuning process for 6000 steps, balancing between sufficient learning and avoiding over-fitting.

f) Computational Resources and Training Environment

The fine-tuning process was conducted using an NVIDIA RTX 3060 GPU with 12GB of VRAM, an Intel Core i7 12700F CPU, and 32GB of RAM. The training spanned 32 hours. The software environment included PEFT version 0.11.1, Transformers version 4.41.2, Pytorch version 2.3.0+cu121, Datasets version 2.19.1, and Tokenizers version 0.19.1. These configurations ensured efficient processing and optimal model performance.

g) Evaluation metrics

Fig. 2 showcase the process to evaluate a model's response, we considered three possible answer formats: (C1) a single digit indicating the correct answer position (e.g., '1'), (C2) the exact text of the correct answer (e.g., 'correct answer') and (C3) the correct answer preceded by '1.' (e.g., '1. correct answer'). For the first type, evaluation is straightforward with a direct comparison to the label field. However, for the second and third formats, we employed the ROUGE metrics to account for potential variations in text representation.

The ROUGE (Recall-Oriented Understudy for Gisting Evaluation) metrics offer a flexible means of comparing textual data based on overlapping units such as n-grams and longest common subsequences. Specifically, we used ROUGE-1, which calculates the overlap of unigrams (single words), ROUGE-2, which assesses the overlap of bigrams (two consecutive words), ROUGE-L, which measures the longest common subsequence between two texts, and ROUGE-Lsum, which is a variant tailored for summarization tasks by comparing sequences across the entire text.

Using ROUGE allows us to capture partial matches between the predicted answer and the ground truth, accommodating slight variations in phrasing or formatting, which may not affect the correctness of the answer. For example, the ground truth might contain two spaces between words due to mistyping (e.g., correct answer), while the model produces a response with only one space (correct answer). In such cases, a strict text-to-text comparison would incorrectly mark the answer as wrong, even though the content is identical. The ROUGE metric, however, evaluates the

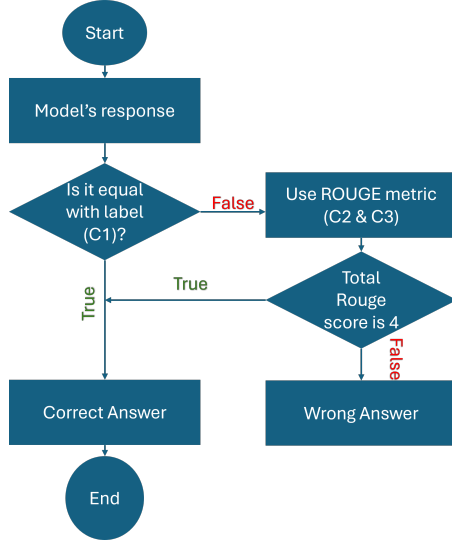


Figure 2. Process to evaluate a model's response

level of overlap between the prediction and ground truth, thereby handling such minor formatting inconsistencies and ensuring that the model is not penalized for trivial errors.

An answer is considered correct if all four ROUGE scores ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-Lsum each yield a perfect score of 1.0. Since each ROUGE metric outputs a value between 0 and 1, a perfect match across all metrics results in a total sum of 4.0. This strict criterion ensures that the model's response matches the ground truth exactly in terms of content, while still being robust to insignificant formatting differences like extra spaces or punctuation variations.

Finally, the model's overall performance was measured using the F1 score. By using the F1 score in the final step, we compared the performance between the Phi-3 models (fine-tuned and not fine-tuned) in this study and other models from Zheng's [19] and Chalkidis's [23] results, which also used the CaseHOLD dataset. This comparison provided a comprehensive measure of the model's accuracy and effectiveness.

III. RESULTS AND DISCUSSION

The results of our study, along with the comparative results from Zheng et al. [19] and Chalkidis et al. [23], are summarized in Table III. The base Phi-3-mini-4k-instruct model, with 3.8 billion parameters, achieved an F1 score of 66.84 without any fine-tuning. This performance is notably higher than the standard BERT model from [19]'s results (61.30) and even the BERT (double) model with twice the training steps (62.30). However, it falls short when compared to more specialized legal models such as Legal-BERT and Custom Legal-BERT, which achieved F1

Table III
EXPERIMENTS'S RESULTS

Model	Params	Training steps	Vocab size	F1 score
Zheng et al's result[19]				
BERT	110M	1M	30K	61.30
BERT (double)	110M	2M	30K	62.30
Legal-BERT	110M	2M	30K	68.00
Custom Legal-BERT	110M	2M	32K	69.50
Chalkidis et al's result[23]				
BERT	110M	1M	32K	70.80
RoBERTa	125M	100K	50K	71.40
DeBERTa	139M	1M	50K	72.60
Longformer	149M	65K	50K	71.90
Bigbird	127M	1M	32K	70.80
Legal-BERT	110M	1M	32K	75.30
CaseLaw-BERT	110M	2M	32K	75.40
Our result				
Phi-3 mini 4K (base)	3.8B	-	32K	66.84
Fine-tuned Phi-3 mini 4K	3.8B	6K	32K	75.76

scores of 68.00 and 69.50, respectively. This is indicative of the base model's general-purpose capabilities, which, without fine-tuning, do not fully leverage the domain-specific nuances required for optimal performance in legal document understanding. Upon fine-tuning with 6,000 training steps, the Phi-3-mini-4k-instruct's performance significantly improved, achieving an F1 score of 75.76. This result surpasses all models in [19] and [23] results, including Legal-BERT (75.30) and CaseLaw-BERT (75.40). This improvement highlights the effectiveness of fine-tuning in enhancing the model's understanding of legal documents and underscores the importance of domain-specific fine-tuning.

A comparative analysis of parameter efficiency and training steps reveals that the base Phi-3-mini-4k-instruct model, with its 3.8 billion parameters, is significantly larger than the models used by [19] and [23], which typically have around 110M to 149M parameters. Despite the larger parameter size, the base model's performance was initially modest, highlighting the necessity of fine-tuning. Fine-tuning with only 6,000 training steps, compared to the millions of steps used by other models, demonstrates the efficiency and rapid adaptability of Phi-3-mini-4k-instruct in acquiring domain-specific knowledge. Several key points emerge from our analysis: the scalability and fine-tuning potential of the Phi-3-mini-4k-instruct model, driven by its large parameter count, enable it to capture complex dependencies and patterns within the data. The dramatic performance improvement post-fine-tuning indicates that while the base model is competent, its true potential is

realized when it is adapted to specific domains through fine-tuning.

An interesting observation emerges when analyzing the impact of vocabulary size across the different studies. Results [19] demonstrated a marginal improvement in F1 score for Legal-BERT when increasing the vocabulary size by 2,000 tokens (from 30K to 32K). However, a similar vocabulary increase for the BERT model in [23] resulted in a substantial performance boost (from 61.3 to 70.8 F1 score). This discrepancy suggests that vocabulary size alone does not dictate performance gains and its impact is likely intertwined with other factors such as training data, model architecture, and fine-tuning methodologies. This observation raises the question of whether a larger vocabulary size, for instance, 50K for Phi-3, could further enhance its performance. Further investigation is needed to disentangle the interplay between vocabulary size and other model parameters for optimizing legal document understanding.

While our findings are encouraging, this study has limitations. Firstly, both the CaseHOLD dataset and the Phi-3 model are English-centric, restricting the generalizability of our conclusions. Future work should explore multilingual legal datasets and models to address this constraint. Secondly, our evaluation focused solely on multiple choice question answering using CaseHOLD. Investigating Phi-3's capabilities on diverse legal tasks and datasets, such as LexGlue, COLIEE or CAIL, would provide a more comprehensive assessment of its strengths and weaknesses. Finally, our focus remained on zero-shot performance and a single model architecture. Exploring few-shot prompting techniques and evaluating other promising open-source models, like LLaMa-3.1 or Gemma 2, could reveal further performance gains and provide valuable insights into the optimal model and training paradigm for legal document understanding.

IV. CONCLUSION

Our study demonstrated the effectiveness of fine-tuning Phi-3-mini-4k-instruct for legal document understanding. The model's performance improvement from an F1 score of 66.84 to 75.76 after fine-tuning highlights its adaptability and potential in specialized domains. This research showcases the capabilities of compact yet powerful language models in tackling complex tasks such as legal text analysis. While our study demonstrated the efficacy of Phi-3-mini-4k-instruct in legal document understanding, several promising avenues for future research emerge. Firstly, exploring the impact of a larger vocabulary size on Phi-3's performance could reveal further optimization potential. Secondly, expanding the evaluation to encompass diverse

legal tasks and datasets, such as those within LexGLUE, COLIEE, or CAIL, would provide a more holistic view of the model's capabilities and generalizability across different legal domains. Thirdly, investigating the effectiveness of few-shot prompting techniques with Phi-3 could uncover new avenues for performance enhancement in resource-constrained scenarios. Lastly, benchmarking against other cutting edge open-source models, including LLaMa-3.1 and Gemma2, would contribute valuable insights into the comparative strengths and weaknesses of different architectures and pre-training paradigms for legal NLP applications.

REFERENCES

- [1] Y. Li, G. Hu, J. Du, H. Abbas, and Y. Zhang, "Multi-task reading for intelligent legal services," *Future Generation Computer Systems*, vol. 113, pp. 218–227, 2020.
- [2] S. Yurii, P. Nataliia, T. Tetiana, P. Tetiana, and H. Stanislav, "Official document as a legal act: Essential aspects," *J. Legal Ethical & Regul. Issues*, vol. 22, p. 1, 2019.
- [3] F. u. Hassan, T. Le, and X. Lv, "Addressing legal and contractual matters in construction using natural language processing: A critical review," *Journal of Construction Engineering and Management*, vol. 147, no. 9, p. 03121004, 2021.
- [4] S. R. Ahmad, D. Harris, and I. Sahibzada, "Understanding legal documents: classification of rhetorical role of sentences using deep learning and natural language processing," in *2020 IEEE 14th International Conference on Semantic Computing (ICSC)*. IEEE, 2020, pp. 464–467.
- [5] G. Marino, D. Licari, P. Bushipaka, G. Comandé, T. Cucinotta *et al.*, "Automatic rhetorical roles classification for legal documents using legal-transformeroverbert," in *CEUR WORKSHOP PROCEEDINGS*, vol. 3441. CEUR-WS, 2023, pp. 28–36.
- [6] I. Timmer and R. Rietveld, "Rule-based systems for decision support and decision-making in dutch legal practice. a brief overview of applications and implications," *Droit et Societe*, vol. 103, pp. 517–534, 11 2019.
- [7] D. Bourcier and G. Clergue, "From a rule-based conception to dynamic patterns. analyzing the self-organization of legal systems," *Artificial Intelligence and Law*, vol. 7, pp. 211–225, 1999.
- [8] K. Pal and J. A. Campbell, "An application of rule-based and case-based reasoning within a single legal knowledge-based system," *ACM SIGMIS Database: the DATABASE for Advances in Information Systems*, vol. 28, pp. 48–63, 9 1997. [Online]. Available: <https://dl.acm.org/doi/10.1145/277339.277344>
- [9] W. Liu, P. Zhou, Z. Zhao, Z. Wang, Q. Ju, H. Deng, and P. Wang, "K-bert: Enabling language representation with knowledge graph," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, pp. 2901–2908, 4 2020. [Online]. Available: <https://ojs.aaai.org/index.php/AAAI/article/view/5681>
- [10] C. Xiao, X. Hu, Z. Liu, C. Tu, and M. Sun, "Lawformer: A pre-trained language model for chinese legal long documents," *AI Open*, vol. 2, pp. 79–84, 2021. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S2666651021000176>
- [11] S. Douka, H. Abdine, M. Vazirgiannis, R. E. Hamdani, and D. R. Amariles, "Juribert: A masked-language model adaptation for french legal text," *Natural Legal Language*

- Processing Workshop 2021*, pp. 95–101, 2021. [Online]. Available: <https://aclanthology.org/2021.nllp-1.9.pdf>
- [12] M. Masala, R. Iacob, A. S. Uban, M.-A. Cidotă, H. Velicu, T. Rebedea, and M. Popescu, “jurbert: A romanian bert model for legal judgement prediction,” *Natural Legal Language Processing Workshop 2021*, pp. 86–94, 2021. [Online]. Available: <https://aclanthology.org/2021.nllp-1.8.pdf>
- [13] H.-T. Nguyen, M.-K. Phi, X.-B. Ngo, V. Tran, L.-M. Nguyen, and M.-P. Tu, “Attentive deep neural networks for legal document retrieval,” *Artificial Intelligence and Law*, vol. 32, pp. 57–86, 3 2024. [Online]. Available: <https://link.springer.com/10.1007/s10506-022-09341-8>
- [14] H. N. Van, D. Nguyen, P. M. Nguyen, and M. Nguyen, “Miko team: Deep learning approach for legal question answering in alqac 2022,” *2022 14th International Conference on Knowledge and Systems Engineering (KSE)*, pp. 1–5, 2022. [Online]. Available: <https://arxiv.org/pdf/2211.02200>
- [15] M. Kejriwal, “Ai in industry today,” in *Artificial Intelligence for Industries of the Future: Beyond Facebook, Amazon, Microsoft and Google*. Springer, 2022, pp. 47–73.
- [16] J. Sauvola, S. Tarkoma, M. Klemettinen, J. Riekk, and D. Doermann, “Future of software development with generative ai,” *Automated Software Engineering*, vol. 31, no. 1, p. 26, 2024.
- [17] X. Han, Z. Zhang, N. Ding, Y. Gu, X. Liu, Y. Huo, J. Qiu, Y. Yao, A. Zhang, L. Zhang *et al.*, “Pre-trained models: Past, present and future,” *AI Open*, vol. 2, pp. 225–250, 2021.
- [18] M. Abidin and S. A. Jacobs, “Phi-3 technical report: A highly capable language model locally on your phone,” 4 2024. [Online]. Available: <http://arxiv.org/abs/2404.14219>
- [19] L. Zheng, N. Guha, B. R. Anderson, P. Henderson, and D. E. Ho, “When does pretraining help?: assessing self-supervised learning for law and the casehold dataset of 53,000+ legal holdings,” in *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Law*. ACM, 6 2021, pp. 159–168.
- [20] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, “Lora: Low-rank adaptation of large language models,” 6 2021. [Online]. Available: <http://arxiv.org/abs/2106.09685>
- [21] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, “Qlora: Efficient finetuning of quantized llms,” in *Advances in Neural Information Processing Systems*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., vol. 36. Curran Associates, Inc., 5 2023, pp. 10 088–10 115. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2023/file/1feb87871436031bdc0f2beaa62a049b-Paper-Conference.pdf
- [22] L. Ouyang, J. Wu, and X. Jiang, “Training language models to follow instructions with human feedback,” in *Advances in Neural Information Processing Systems*, vol. 35. Curran Associates, Inc., 3 2022, pp. 27 730–27 744. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf
- [23] I. Chalkidis, A. Jana, D. Hartung, M. Bommarito, I. Androustopoulos, D. M. Katz, and N. Aletras, “Lexglue: A benchmark dataset for legal language understanding in english,” *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, vol. 1, pp. 4310–4330, 10 2022. [Online]. Available: <https://arxiv.org/abs/2110.00976v4>



Email: khanhnh@tnu.edu.vn

Nguyen Huu Khanh has graduated with a Master’s degree in Computer Science from the University of Information and Communications Technology - Thai Nguyen University since 2022 and is currently a PhD student here since 2023. His main research interests are Computer Science, Natural Language Processing and Computer Vision.



Nguyen Van Viet received the Bachelor’s Information Technology at Thai Nguyen University in 2009 and Master’s degree Information Technology at Manuel S. Enverga University Foundation, Lucena City, Philippines in 2012. He worked as a lecturer at the Faculty of Information Technology, School of Information and Communication Technology, Thai Nguyen University from 2009. Now, he is a researcher at the Thai Nguyen University of Information and Communication Technology, Thai Nguyen, Vietnam. Office address: University of Information and Communication Technology, Thai Nguyen University, Thai Nguyen, Vietnam.
Email: nviet@ictu.edu.vn

Nguyen Van Viet received the Bachelor’s Information Technology at Thai Nguyen University in 2009 and Master’s degree Information Technology at Manuel S. Enverga University Foundation, Lucena City, Philippines in 2012. He worked as a lecturer at the Faculty of Information Technology, School of Information and Communication Technology, Thai Nguyen University from 2009. Now, he is a researcher at the Thai Nguyen University of Information and Communication Technology, Thai Nguyen, Vietnam. Office address: University of Information and Communication Technology, Thai Nguyen University, Thai Nguyen, Vietnam.
Email: nviet@ictu.edu.vn



Email: vinhnt@ictu.edu.vn

Nguyen The Vinh received his PhD in Computer Science from Texas Tech University in 2020. He is currently a lecturer in Software Engineering at the University of Information and Communication Technology - Thai Nguyen University. His main research interests are Computer Science and AI.



Nguyen Huu Cong received his PhD in automatic control from Hanoi University of Science and Technology in 2003 and was promoted to Associate Professor in 2007. His main research interests are automatic control and optimal control for objects with distributed and slowly changing parameters.
Email: conghn@tnu.edu.vn

Cold Storage Temperature Monitoring with Trilinear Interpolation

Minh Thai Truong, Minh Bang Le, Cong Danh Cao, Phuong Lan Phan, Huynh Tram Vo
Can Tho University, Can Tho, Vietnam

Correspondence: Minh Thai Truong, tmthai@cit.ctu.edu.vn. Minh Bang Le, lmbang@cit.ctu.edu.vn

Received: 14/07/2024, revised: 20/09/2024, accepted: 11/10/2024

Digital Object Identifier: 10.32913/mic-ict-research-vn.v2024.n3.1317

Abstract: The Internet of Things (IoT) is widely applied in agriculture and industry. This article proposes a 3-dimensional temperature monitoring model for cold storage, enhancing the ability to observe and detect abnormal temperature changes within the storage. Using real-time temperature acquisition and applying trilinear interpolation techniques, this model determines temperatures at all locations in the storage. It accepts temperature values and sensor locations, appropriately segmenting the cold storage space to apply trilinear interpolation and determine the temperature value at every location in the storage. The model was tested based on different number of sensors and different sensor locations. Based on that, we provide an optimal solution for the number of sensors needed for cold storage.

Keywords: *lot, cold storage, temperature, trilinear interpolation, monitor.*

Tiêu đề: Giám sát nhiệt độ kho lạnh bằng phương pháp nội suy ba chiều

Tóm tắt: Internet vạn vật (IoT) được ứng dụng rộng rãi trong nông nghiệp và công nghiệp. Bài báo này đề xuất một mô hình giám sát nhiệt độ ba chiều cho kho lạnh, tăng cường khả năng quan sát và phát hiện những thay đổi nhiệt độ bất thường trong kho. Sử dụng việc thu thập nhiệt độ theo thời gian thực và áp dụng các kỹ thuật nội suy tam tuyến, mô hình này xác định nhiệt độ tại tất cả các vị trí trong kho. Nó chấp nhận các giá trị nhiệt độ và vị trí cảm biến, phân đoạn không gian kho lạnh một cách thích hợp để áp dụng nội suy ba tuyến tính và xác định giá trị nhiệt độ tại mọi vị trí trong kho. Mô hình đã được thử nghiệm dựa trên số lượng cảm biến khác nhau và các vị trí cảm biến khác nhau. Dựa trên đó, chúng tôi cung cấp một giải pháp tối ưu cho số lượng cảm biến cần thiết cho kho lạnh.

Từ khóa: *lot, cold storage, temperature, trilinear interpolation, monitor.*

I. INTRODUCTION

IoT applications are widely applied in agriculture and industry [1]. This article focuses on improving IoT usage in cold storage temperature management. Specifically, using temperature sensors in cold storage allows managers to monitor real-time temperature changes, providing timely solutions to ensure product quality is maintained optimally.

However, applying temperature sensors in cold storage has limitations. These include managing the number of sensors, sensor installation locations, and accuracy when analyzing data. One significant challenge is determining the optimal number of sensors to ensure effective monitoring while minimizing costs. Sensor installation locations significantly impact monitoring efficacy, and the accuracy of temperature data is a concern when sensors cannot cover all required locations.

To address these issues, we propose a model that calculates all temperature values within the cold storage and dis-

plays a 3-dimensional (3D) simulation of these values on a web platform, called Cold Storage Temperature Simulation with Trilinear Interpolation. The goal is to support visual monitoring and plan appropriate sensor installations for maximum efficiency and cost savings. The model calculates temperature values at all locations within the cold storage using trilinear interpolation, even in areas without sensors. It relies on three factors: storage size for sensor location estimation, sensor placement for determining storage division before temperature calculation, and real-time temperature values from sensors installed within the storage.

The content of this paper is arranged into six main parts. The first section is the introduction, which provides a general presentation of the contents of the study. The second section is the related work that presents previous studies in the same field. The third section is the proposed model that provides a detailed description of the research problem-solving method. The fourth section is the exper-

iment that provides detailed test scenarios based on the proposed model. The fifth section is the conclusion, which summarizes the problems the study has solved and future development directions. The final section is the references that provide annotate the documents used for reference in the study.

II. RELATED WORK

1. Trilinear Interpolation

Trilinear interpolation is a 3D extension of linear and bilinear interpolation. It estimates values within a cube by performing linear interpolation along each axis based on the values at the cube's eight corners.

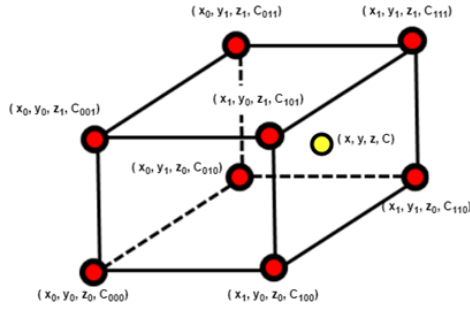


Figure 1. The 3-dimensional space used for interpolation

The Trilinear interpolation function gives results at any point (x, y, z) according to formula (1):

$$f(x, y, z) = a_0 + a_1x + a_2y + a_3z + a_4xy + a_5xz + a_6yz + a_7xyz \quad (1)$$

The values at the vertices are usually known, so we need to compute the coefficients $\{a_0, a_1, a_2, \dots, a_7\}$. These coefficients are then used in the interpolation function's linear equation.

$$\begin{bmatrix} 1 & x_0 & y_0 & z_0 & x_0y_0 & x_0z_0 & y_0z_0 & x_0y_0z_0 \\ 1 & x_1 & y_0 & z_0 & x_1y_0 & x_1z_0 & y_1z_0 & x_1y_0z_0 \\ 1 & x_0 & y_1 & z_0 & x_0y_1 & x_0z_1 & y_0z_1 & x_0y_1z_0 \\ 1 & x_1 & y_1 & z_0 & x_1y_1 & x_1z_1 & y_1z_1 & x_1y_1z_0 \\ 1 & x_0 & y_0 & z_1 & x_0y_0 & x_0z_0 & y_0z_0 & x_0y_0z_1 \\ 1 & x_1 & y_0 & z_1 & x_1y_0 & x_1z_0 & y_1z_0 & x_1y_0z_1 \\ 1 & x_0 & y_1 & z_1 & x_0y_1 & x_0z_1 & y_0z_1 & x_0y_1z_1 \\ 1 & x_1 & y_1 & z_1 & x_1y_1 & x_1z_1 & y_1z_1 & x_1y_1z_1 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \end{bmatrix} = \begin{bmatrix} c_{000} \\ c_{100} \\ c_{010} \\ c_{110} \\ c_{001} \\ c_{101} \\ c_{011} \\ c_{111} \end{bmatrix} \quad (2)$$

Solving equation (2) we get the correlation coefficients $\{a_1, a_2, \dots, a_7\}$ as in formula (3):

$$\begin{aligned} a_0 &= \frac{-c_{000}x_1y_1z_1 + c_{001}x_1y_1z_0 + c_{010}x_1y_0z_1 - c_{011}x_1y_0z_0}{(x_0 - x_1)(y_0 - y_1)(z_0 - z_1)} \\ &\quad + \frac{c_{100}x_0y_1z_1 - c_{101}x_0y_1z_0 - c_{110}x_0y_0z_1 + c_{111}x_0y_0z_0}{(x_0 - x_1)(y_0 - y_1)(z_0 - z_1)} \\ a_1 &= \frac{c_{000}y_1z_1 - c_{001}y_1z_0 - c_{010}y_0z_1 + c_{011}y_0z_0}{(x_0 - x_1)(y_0 - y_1)(z_0 - z_1)} \\ &\quad - \frac{c_{100}y_1z_1 + c_{101}y_1z_0 + c_{110}y_0z_1 - c_{111}y_0z_0}{(x_0 - x_1)(y_0 - y_1)(z_0 - z_1)} \\ a_2 &= \frac{c_{000}x_1z_1 - c_{001}x_1z_0 - c_{010}x_1z_1 + c_{011}x_1z_0}{(x_0 - x_1)(y_0 - y_1)(z_0 - z_1)} \\ &\quad + \frac{-c_{100}x_0z_1 + c_{101}x_0z_0 + c_{110}x_0z_1 - c_{111}x_0z_0}{(x_0 - x_1)(y_0 - y_1)(z_0 - z_1)} \\ a_3 &= \frac{c_{000}x_1y_1 - c_{001}x_1y_1 - c_{010}x_1y_0 + c_{011}x_1y_0}{(x_0 - x_1)(y_0 - y_1)(z_0 - z_1)} \\ &\quad + \frac{-c_{100}x_0y_1 + c_{101}x_0y_1 + c_{110}x_0y_0 - c_{111}x_0y_0}{(x_0 - x_1)(y_0 - y_1)(z_0 - z_1)} \\ a_4 &= \frac{-c_{000}z_1 + c_{001}z_0 + c_{010}z_1 - c_{011}z_1}{(x_0 - x_1)(y_0 - y_1)(z_0 - z_1)} \\ &\quad + \frac{c_{100}z_1 - c_{101}z_0 - c_{110}z_1 + c_{111}z_0}{(x_0 - x_1)(y_0 - y_1)(z_0 - z_1)} \\ a_5 &= \frac{-c_{000}y_1 + c_{001}y_1 + c_{010}y_0 - c_{011}y_0}{(x_0 - x_1)(y_0 - y_1)(z_0 - z_1)} \\ &\quad + \frac{c_{100}y_1 - c_{101}y_1 - c_{110}y_0 + c_{111}y_0}{(x_0 - x_1)(y_0 - y_1)(z_0 - z_1)} \\ a_6 &= \frac{-c_{000}x_1 + c_{001}x_1 + c_{010}y_1 - c_{011}x_1}{(x_0 - x_1)(y_0 - y_1)(z_0 - z_1)} \\ &\quad + \frac{c_{100}x_0 - c_{101}x_0 - c_{110}x_0 + c_{111}x_0}{(x_0 - x_1)(y_0 - y_1)(z_0 - z_1)} \\ a_7 &= \frac{c_{000} - c_{001} - c_{010} + c_{011} - c_{100} + c_{101} + c_{110} - c_{111}}{(x_0 - x_1)(y_0 - y_1)(z_0 - z_1)} \end{aligned} \quad (3)$$

In short, applying formulas (1), (2), (3), we can calculate the entire set of coefficients $\{a_1, a_2, \dots, a_7\}$ as shown above to determine any value inside the space of the given rectangular box.

2. IoT-Based Cold Storage Monitoring and Controlling Systems

Patel et al. [2] introduced a comprehensive Smart Cold Storage and Inventory Monitoring System that leverages IoT technology to significantly improve the efficiency and reliability of cold storage management. Their system integrates real-time monitoring capabilities with early warning alerts, providing complete end-to-end visibility throughout the entire product value chain. This integration allows for precise control of temperature and inventory levels, which is crucial for maintaining the quality and safety

of stored goods. However, the system's high complexity and substantial implementation costs may present barriers, particularly for smaller operations with limited budgets. Furthermore, the system's dependence on continuous internet connectivity could be a critical vulnerability in regions experiencing unstable network access, potentially impacting the reliability of the monitoring and control functions.

In contrast, Chen et al. [3] focused on developing an IoT-Based Low-Cost Cold Storage Atmosphere Monitoring and Controlling System, aiming to offer a more affordable solution while still providing essential monitoring and control functions. Their system utilizes a combination of temperature and humidity sensors, a microcontroller, and a mobile application to deliver real-time updates and allow for manual adjustments in cold storage environments. This approach is designed to be cost-effective, making it accessible for facilities with budget constraints, and helps in maintaining optimal storage conditions for perishable goods. However, this low-cost model may impact the system's overall durability and accuracy, as the sensors used may not be as robust or precise over time compared to more expensive alternatives. Additionally, the system's limited scalability could pose challenges for larger storage facilities that require more extensive monitoring capabilities.

III. PROPOSED MODEL

This section proposes the "Cold Storage Temperature Monitoring with Trilinear Interpolation" model (CSTM) for calculating and monitoring temperature parameters in cold storage using storage size, sensor locations, and real-time sensor temperature data. To compare with the research [2], our proposed model uses interpolation to determine temperatures at locations without sensors, thereby reducing the number of sensors needed and lowering costs, while still providing necessary warnings. In addition, our research improves upon the research [3] by demonstrating efficient sensor placement, optimizing the number of sensors needed, and ensuring cost savings without compromising accuracy.

1. CSTM Overview

In this study, we conduct two main stages:

Firstly, based on sensor numbers, we divide the cold storage space, ensuring compliance with trilinear interpolation techniques and using all sensor values inside the storage. This stage outputs divided spaces within the storage.

Secondly, once the storage space is divided, we perform trilinear interpolation to calculate temperature values at virtual sensors, using real and virtual sensor values to interpolate temperatures within all divided spaces. Finally,

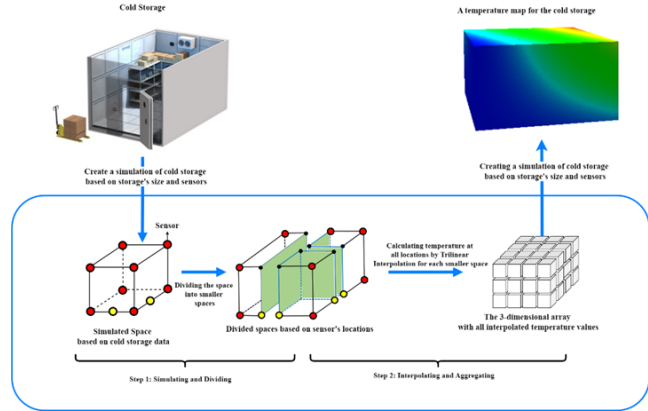


Figure 2. Overview of CSTM

we aggregate these values to form a comprehensive temperature picture of the entire storage.

Finally, we change the location and number of sensors to create different experiments. From there, the difference can be evaluated, and improvement methods can be drawn to help optimize the use of sensors.

2. Simulating and Dividing

In this phase, we determine the number of sensors in the cold storage, which is the most critical factor in dividing the cold storage space. Fig.3 is the flowchart of the algorithm used for dividing and saving all storage spaces.

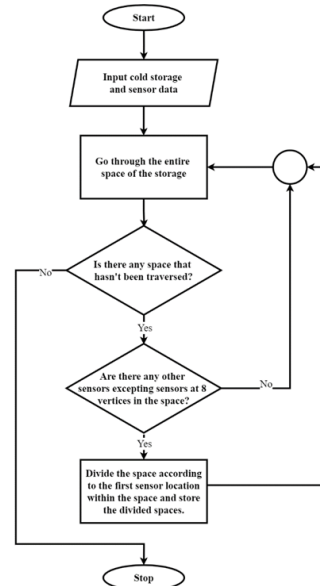


Figure 3. The algorithm divides spaces of the storage

Based on the number of sensors, we divide into 3 cases that need to be handled as follows:

Case 1: There are only eight sensors at eight corners of the cold storage, corresponding to the requirements of the trilinear interpolation technique. In this case, we propose to skip the space division step because the cold storage space already meets the technical conditions and real-time interpolation can be performed anytime the sensors update the price new treatment.

Case 2: There are nine sensors inside the cold storage, including eight sensors at the eight roots (vertices of a space or the storage) of the storage and one outside of those locations. In this case, we propose a solution reference table based on the location of that ninth sensor (x, y, z) as follows:

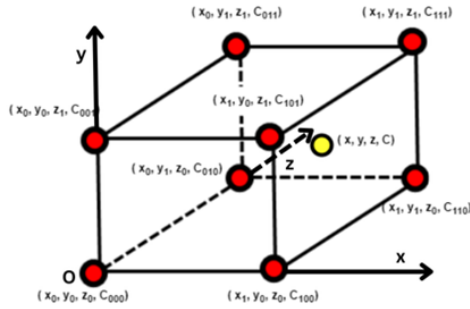


Figure 4. Eight sensors at eight vertices of the storage

Table I
DIVIDING SPACE BASED ON AN ADDED SENSOR
LOCATION – THE 9TH SENSOR'S LOCATION

Situation	Ox axis	Oy axis	Oz axis	Number of separated spaces
Inside	$x_0 < x < x_1$	$y_0 < y < y_1$	$z_0 < z < z_1$	8
On face	$x_0 = x \parallel x = x_1$	$y_0 < y < y_1$	$z_0 < z < z_1$	4
	$x_0 < x < x_1$	$y_0 = y \parallel y = y_1$	$z_0 < z < z_1$	
	$x_0 < x < x_1$	$y_0 < y < y_1$	$z_0 = z \parallel z = z_1$	
On edge	$x_0 < x < x_1$	$y_0 = y \parallel y = y_1$	$z_0 = z \parallel z = z_1$	2
	$x_0 = x \parallel x = x_1$	$y_0 < y < y_1$	$z_0 = z \parallel z = z_1$	
	$x_0 = x \parallel x = x_1$	$y_0 = y \parallel y = y_1$	$z_0 < z < z_1$	

In there:

- x_0 (x_1): is the smallest (largest) x value on the Ox axis.
- y_0 (y_1): is the smallest (largest) y value on the Oy axis.
- z_0 (z_1): is the smallest (largest) z value on the Oz axis.

Each situation is reproduced graphically as follows:

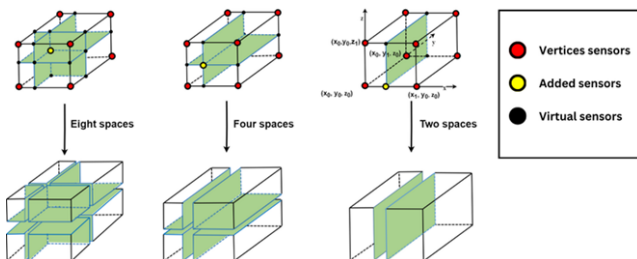


Figure 5. The results of divided spaces based on sensor locations

During the division process, roots with no sensors will be marked as virtual sensors. Their temperatures will be

updated in the interpolation step using the eight original sensors of the cold storage. The virtual sensor will have the same role as the sensor at eight roots.

Case 3: More than nine sensors were added to the cold storage space. In this case, we propose that for each new sensor (other than the original eight sensors) added, the space containing that new sensor will be redistributed according to the convention of the 9-sensor case in cold storage. This makes us immediately think of recursive techniques in programming, and its stopping condition is when a space only has eight sensors left at eight roots.

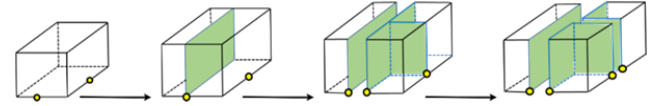


Figure 6. Dividing space process based on additional sensor locations

3. Interpolating and Aggregating

Applying the interpolation technique to the problem of calculating temperature inside cold storage, we can set some prerequisites as follows:

- Cold storage's spaces must be rectangular.
- Eight sensors must be installed at the corners of the storage or the space, which correlates with the available values in the example above.

From there, we can calculate all the values inside the cold storage based on a set of coefficients deduced from the temperature values of 8 sensors at eight corners. In particular, temperature values will be calculated in real-time as soon as the sensors update the values.

In addition, we also built an algorithm to support the use of trilinear interpolation in cases where there were more than eight original sensors as the Fig. 7.

This means we can add other sensors to the cold storage at any location and use them for calculations without violating the conditions of the trilinear interpolation technique.

During this phase, we execute two main jobs and these jobs are executed periodically in real time to ensure accuracy in temperature monitoring of cold storage.

First, we perform trilinear interpolation based on 8 sensors of the cold storage. From there, we determined the temperature values at the virtual sensors that were previously determined in the previous stage.

Second, based on the values at the 8 vertices of each space, it has been divided to perform the trilinear interpolation technique of all temperature values in that space. Finally, we aggregate all the divided spaces to form a temperature picture of the entire cold storage.

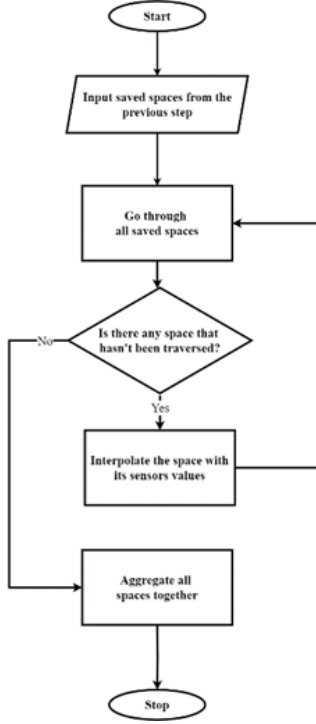


Figure 7. The algorithm interpolates and aggregates spaces of the storage



Figure 8. The aggregated space

As a result, all real-time cold storage temperature values have been completely calculated.

4. Evaluating the proposed model

Using a cold storage size (length, width, height), position (x, y, z in the Oxyz coordinate axis) and current temperature ($^{\circ}\text{C}$) at all sensors in that cold storage as the input, the proposed model gives the main result, which is an array denoted as (4):

$$T = \{(x, y, z, C_{xyz}) | 0 \leq x < a, 0 \leq y < b, 0 \leq z < c\} \quad (4)$$

In there, each array will have $N = a * b * c$ elements, where a = cold storage length + 1, b = cold storage width + 1, c = cold storage height + 1 because we consider coordinates from position 0 to the end of a certain length.

From there, we change sensor locations and the number of sensors for our experiments differently. Then, we

measure their difference by the average deviation. This is necessary to help optimize the cost and the accuracy of results by finding which locations sensors can be placed and the number of sensors as an improvement method. For the 3D array result of each experiment, we call them T_i ($i = 1, 2, \dots, n$ where n is the number of experiments). To do that, we follow a series of steps below:

We calculate the absolute deviation between any two of the above those arrays according to the following formula (5):

$$\Delta_{ij} = |T_i - T_j| = \{e_k, k \in [1; N]\} \quad (5)$$

In there: Δ_{ij} is a 3D array that stores the absolute deviation of each pair of values in the two arrays. Based on that, the absolute deviation between spaces of i and j is Δ_{ij} .

Then, we perform the sum of all values inside each absolute deviation array, and the result is as a following formula (6):

$$S_{ij} = \sum_{k=1}^N e_k, e_k \in \Delta_{ij} \quad (6)$$

Next, we apply the formula (7) to calculate the average deviation for each pair of arrays:

$$D_{ij} = \frac{S_{ij}}{N} \quad (7)$$

In there:

- D_{ij} is the average deviation for each pair of arrays.
- S_{ij} is the sum of the absolute deviations of all elements in an array.
- N is the total number of elements in an array ($N = a * b * c$).

We can evaluate the influence of the number of divided spaces based on the average difference between the resulting arrays. From there, we can find a way to set the sensor to increase or decrease the number of divided spaces to reduce that difference.

IV. EXPERIMENT

1. Install sensors

In this study, we installed a thermal sensor system based on cold storage. According to recommendation of [4] the cold storage should have 12 meters in length, 3 meters in width, and 3 meters in height with a temperature range from -12°C to -8°C as a suitable temperature to maintain frozen products. Besides, we conducted three experiments as follows:

- Scenario 1: Install eight sensors at eight corners of the cold storage to capture a particular space and perform interpolation based on that space.

- Scenario 2: Install the 9th sensor to check the spatial division according to the output model. Then, interpolation will be performed based on the divided spaces.
- Scenario 3: Install the 10th sensor to check the spatial division according to the output model recursively.

Based on the evaluation of the average deviation between the spaces, we evaluated the above 3 scenarios and proposed an optimization direction in sensor installation.

2. Scenario 1: Install eight sensors at eight vertices

We build a sensor network consisting of 8 sensors at eight roots of the cold storage. Based on the division method, the cold storage will be divided into a single space because there are no sensors other than the eight sensors at the eight roots attached.

Below is information about the original eight sensors; the sensor values will change in real-time:

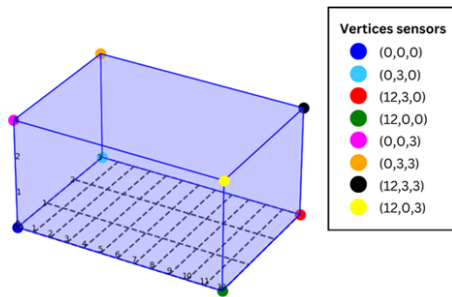


Figure 9. Sensor locations of scenario 1

Below are the resulting parameters after performing trilinear interpolation for the entire cold storage based on the number and changes in temperature values of those sensors.

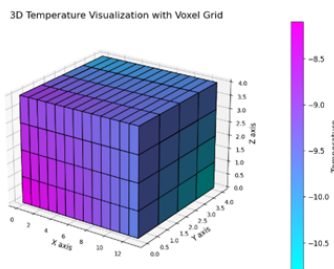


Figure 10. The result of scenario 1

3. Scenario 2: Install nine sensors

We add cold storage on a sensor at any location and check the number of divided spaces. This helps verify the model's

ability to divide space according to sensor changes. In this case, we add a sensor at (8, 0, 0). Based on the number and location of sensors, our model splits the space into 2 smaller spaces including:

Table II
DIVIDED SPACES OF SCENARIO 2

Space No	x0	y0	z0	x1	y1	z1
1	0	0	0	8	3	3
2	8	0	0	12	3	3

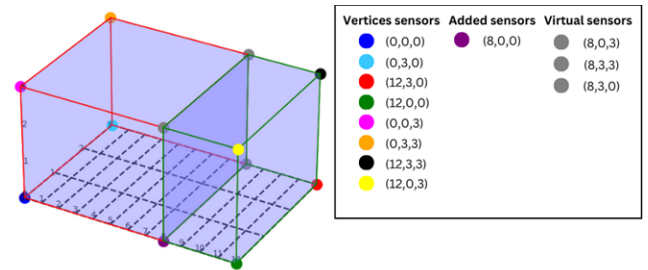


Figure 11. Divided space of scenario 2

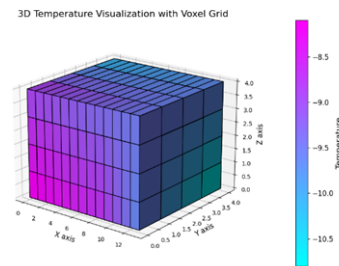


Figure 12. The result of scenario 2

4. Scenario 3: Install ten sensors

We perform additional cold storage on more than one sensor at any location. In this case, we add the 10th sensor at (2, 2, 0). Based on the number and location of sensors, our model splits the space into 5 smaller spaces including:

Table III
DIVIDED SPACES OF SCENARIO 3

Space No	x0	y0	z0	x1	y1	z1
1	0	0	0	2	2	3
2	2	0	0	8	2	3
3	0	2	0	2	3	3
4	2	2	0	8	3	3
5	8	0	0	12	3	3

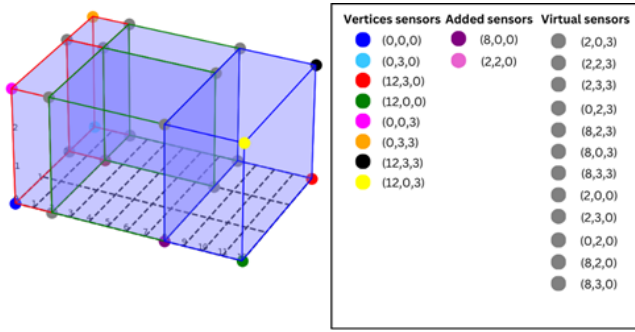


Figure 13. Divided spaces of scenario 3

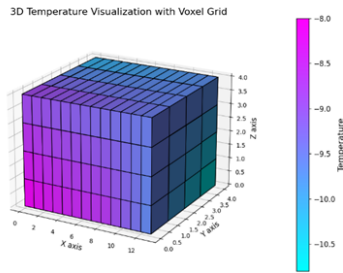


Figure 14. The result of scenario 3

5. Improvement

Based on the above experiments, we realized that the more space is divided, the more virtual sensor values are used to perform interpolation. This can lead to apparent differences between experiments. However, the temperature values at the eight original sensors of each experiment are equal.

To show the difference between experiments, we determine the average deviation of all three experiments as follows:

We call the three 3D arrays that store temperatures in the above three experiments: T1, T2, and T3. Each array has a size of 13x4x4 elements.

We calculate the absolute deviation between any two of the above three arrays. The results:

- The absolute deviation between arrays 1 and 2 is Δ_{12}
- The absolute deviation between arrays 1 and 3 is Δ_{13}
- The absolute deviation between arrays 2 and 3 is Δ_{23}

We perform the sum of all values inside each absolute deviation array, and the result is as follows:

- The total absolute deviation between arrays 1 and 2 is $S_{12} = 33.60000000000005$
- The total absolute deviation between arrays 1 and 3 is $S_{13} = 64.26666666666671$

- The total absolute deviation between arrays 2 and 3 is $S_{23} = 43.26666666666675$

We apply the formula to calculate the average deviation for each pair of arrays (T1, T2), (T1, T3) and (T2, T3). The results:

- The average deviation between arrays 1 and 2 is $D_{12} = 0.1615384615384618$
- The average deviation between arrays 1 and 3 is $D_{13} = 0.30897435897435915$
- The average deviation between arrays 2 and 3 is $D_{23} = 0.2080128205128209$

The above results show that the average deviation reflects the difference in temperature between measurements. The higher the deviation value, the more significant the difference. In this case, the largest deviation between spaces 1 and 3 indicates a significant temperature fluctuation. The deviation between spaces 1 and 2 is the smallest, indicating that the temperature in this area is relatively stable. From there, we can see that the division of space and the use of virtual sensors partly affect the change in temperature after interpolation because the virtual temperature values will be used as a parameter of the trilinear interpolation process. Therefore, we propose an optimal way to install sensors to eliminate virtual sensor values inside the cold storage. We propose a method to reduce virtual sensors in cold storage by strategically placing real sensors to improve temperature interpolation accuracy. This is done by identifying virtual sensor locations created by new sensor installations and replacing them with real sensors.

V. CONCLUSION

This study proposes a model to monitor temperature in the cold storage, clearly showing each temperature position inside the cold storage, even where no temperature sensors are installed. In addition, the study also proposed a solution to optimize the location and number of sensors used through experiments. In the future, this model will be applied in practice to help measure and monitor temperature in cold storages. Furthermore, the model will also be improved to suggest sensor mounting locations to save the number of sensors but achieve the highest efficiency.

ACKNOWLEDGEMENT

This study was funded by the Technical Cooperation Project of the Japan International Cooperation Agency (JICA).

REFERENCES

- [1] Lee, I., & Lee, K., "The Internet of Things (IoT): Applications, investments, and challenges for enterprises", *Business Horizons*, 58(4), 431-440, 2015. <https://doi.org/10.1016/j.bushor.2015.01.007>.

- [2] Patel, K., McKay, T., Wright, T., & Witter, S, "Smart Cold Storage and Inventory Monitoring System", *IEEE Xplore*, 2020. <https://ieeexplore.ieee.org/document/9022815>.
- [3] Chen, Y., Liu, Y., & Zhang, Z, "IoT-Based Low-Cost Cold Storage Atmosphere Monitoring and Controlling System", *IEEE Xplore*, 2018. <https://ieeexplore.ieee.org/document/8456768>.
- [4] U.S. Department of Agriculture (USDA), Food Safety and Inspection Service (FSIS). Freezing and food safety, 2013. <https://www.fsis.usda.gov/wps/portal/fsis/topics/food-safety-education/get-answers/food-safety-basics/freezing-and-food-safety>.



Minh Thai Truong obtained a Master's in Information System Development from HAN University (2010), Netherlands, he obtained his Ph.D in Computer Science from Toulouse 1 Capitole University (2015), France. His research interests include Data Warehousing, Multi-Agent-Based Systems, and the Internet of Things (IoT), applying these technologies to enhance software development in the fields of agriculture and aquaculture.

Email: tmthai@cit.ctu.edu.vn



Minh Bang Le received his Bachelor of Engineering in Software Engineering from Can Tho University, Vietnam (2023). Currently, he is studying for a Master's degree in Computer Science at the National Yang Ming Chiao Tung University, Taiwan.

Email: imbang@cit.ctu.edu.vn



Cong Danh Cao received his Bachelor of Engineering in Software Engineering from Can Tho University, Vietnam (2023). Currently, he is studying for a Master's degree in Artificial Intelligence at the National Central University, Taiwan.

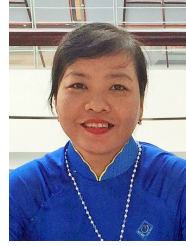
Email: ccdanh@cit.ctu.edu.vn



Phuong Lan Phan obtained a Master's degree at Asian Institute of Technology, Thailand (2003). She obtained her Ph.D in Computer Science from the University of Science and Technology - The University of Da Nang, Vietnam (2020). Now, she is a lecturer in the Faculty of Software Engineering. Her research interests include

Software Processes, and Recommendation Systems.

Email: pplan@cit.ctu.edu.vn



Huynh Tram Vo received her Bachelor of Engineering in Computer Science from Can Tho University (1996). She obtained her Master's degree in Computer Science from Vrije Universiteit Brussel, Belgium (2000). Now, she is a lecturer in the Faculty of Software Engineering at the College of Information and Communication Technology, Can Tho University, Vietnam. Her research interests include the complexity of the algorithms, and Artificial Intelligence.

Email: vhtram@cit.ctu.edu.vn

An Extended Algorithm using a Divide-and-Rule Approach for Boundary Value Analysis and Equivalence Class Partitioning

Danh Nguyen-Cong

Faculty of Software Engineering, College of Information and Communication Technology, Can Tho University, Can Tho, Vietnam

Correspondence: Danh Nguyen-Cong, ncdanh@cit.ctu.edu.vn

Received: 31/07/2024, revised: 23/09/2024, accepted: 11/10/2024

Digital Object Identifier: 10.32913/mic-ict-research-vn.v2024.n3.1311

Abstract: An extended algorithm using a divide-and-rule approach is proposed in this paper. Instead of generating test cases by only using Boundary Value Analysis and an experience-based technique for combination of input variables with functional dependency like the original algorithm, the extended algorithm applies both Boundary Value Analysis and Equivalence Class Partitioning techniques as well as various techniques for combination of input values. In addition, a new tool provides advantages when compared with the original one. It provides users with more choices in generating test cases.

Keywords: *Software testing, boundary value analysis, equivalence class partitioning, functional dependency*

Tiêu đề: Một thuật toán mở rộng sử dụng phương pháp chia để trị cho kỹ thuật phân tích giá trị biên và phân vùng tương đương

Tóm tắt: Bài báo này đề xuất một thuật toán mở rộng sử dụng phương pháp chia để trị. Thay vì tạo các trường hợp kiểm thử chỉ bằng cách sử dụng kỹ thuật Phân tích giá trị biên và kỹ thuật dựa trên kinh nghiệm để kết hợp các biến đầu vào có phụ thuộc hàm như thuật toán gốc, thuật toán mở rộng áp dụng cả kỹ thuật Phân tích giá trị biên và Phân vùng tương đương cũng như nhiều kỹ thuật kết hợp khác để kết hợp các giá trị đầu vào. Ngoài ra, một công cụ mới cung cấp nhiều ưu điểm hơn khi so sánh với công cụ gốc. Nó cung cấp cho người dùng nhiều lựa chọn hơn trong việc tạo các trường hợp thử nghiệm.

Từ khóa: *Software testing, boundary value analysis, equivalence class partitioning, functional dependency*

I. INTRODUCTION

Generating effective test cases to thoroughly evaluate the functionality of developing software is a challenging aspect of the software testing process. Numerous functional testing methods are employed for this purpose [1–4, 6]. Among the various techniques used in black-box testing, Equivalence Class Partitioning (ECP) and Boundary Value Analysis (BVA) are the most prevalent techniques for designing test cases [5, 7, 8]. These techniques are also widely adopted by various software testing tools to automate the creation and evaluation of test cases.

ECP is a software testing technique that divides input data into equivalent partitions to derive test cases. Each partition is tested at least once to uncover error classes, reducing the overall number of test cases and thus shortening

the testing time [9].

BVA is a software testing technique that focuses on input values at edges of their valid ranges. The assumption is that errors are more likely to occur near these extreme values [4, 5]. This method involves testing input values at the minimum, slightly above the minimum, a typical value, slightly below the maximum, and the maximum. These values can be denoted as min, min+, nom, max-, and max, respectively.

However, both ECP and BVA are effective for programs with independent variables but fail when variables are interdependent. For instance, in the NextDate function, the variables Month and Day range from 1-12 and 1-31, respectively. Consequently, using ECP and BVA does not generate test cases for dates like February 28th and 29th.

Vij K. et al. [10] developed a testing tool based on an algorithm that uses a divide-and-rule approach to eliminate dependencies among variables, creating independent variable sets. The algorithm showed the advantages over the traditional boundary value analysis technique.

In this paper, a new testing tool is developed based on a new algorithm that extends the algorithm proposed in [10]. Our extended algorithm considers the integrated use of BVA, ECP and different techniques for combination of input values. Section II presents the divide-and-rule approach including the extended algorithm and an example. Section III introduces the implementation of a new testing tool. The demonstration of the new testing tool is given in Section IV. Finally, Section V summarizes the paper with a conclusion and our future plans.

II. EXTENDED DIVIDE-AND-RULE APPROACH

The divide-and-rule approach was presented in detail in [10]. The algorithm breaks the dependencies among variables and produces multiple independent sets of variables. It ensures that variables within these new sets do not affect each other's boundary values. This allows for traditional boundary value analysis to be performed on each independent set.

In a set of interdependent variables, boundaries of a variable are influenced by the values of other variables, a relationship known as boundary dependency. For the divide-and-rule approach, these variables are categorized into the following three types: 1) *dependent variables*: variables whose boundary values are affected by the values of other variables, 2) *boundary-determining variables*: variables that influence the boundary values of other variables, and 3) *independent variables*: variables that neither affect nor are affected by the boundary values of other variables.

1. Extended Algorithm

The algorithm proposed by Vij K. et al. [10] can create test cases by only using BVA and an experience-based technique for combination of input variables. To provide flexibility and allow users to design effective test cases, we made some changes to this algorithm. Our extended algorithm can apply BVA, ECP and some various techniques for combination of input values. More particularly, we extend Steps 3, 4, 6, 8, 9 and 10 in the algorithm proposed in [10]. All the steps are presented in detail below.

Step 1: Categorize variables into three sets:

- D : set of dependent variables,
- B : set of boundary-determining variables,
- I : set of independent variables.

Step 2: For each dependent variable $d \in D$, create a set $V_{B,d}$ of its determining variables. Every element in $V_{B,d}$ will also belong to set B . The boundary values of d are influenced by all elements in $V_{B,d}$.

Step 3: For each dependent variable $d \in D$, create distinct sets representing all possible boundary value ranges for d . Label these sets d_1 through d_x . If BVA is used, these sets will include the boundary values (min, min+, nom, max-, max). On the other hand, if ECP is used, these sets will only contain the nominal values (nom). Creating these sets in such a way that d can assume, based on its boundary-determining variables.

Step 4: For each boundary-determining variable $b \in B$, create sets of values that impact the boundary values of its dependent variable. These sets are labeled b_1 to b_y . It is important to note that if BVA is applied, sets of values used for each boundary-determining variable b will include the boundary values (min, min+, nom, max-, max). In contrast, the sets of values used for b will only contain the nominal values (nom) when ECP is applied. The relationship between boundary-value sets (d_x) and boundary-determining sets (b_y) is such that if b takes a value from b_y , then d can only take a value from the corresponding d_x . The details of these relationships will be outlined in Step 6.

Step 5: For each dependent variable $d \in D$, create all possible combinations of its boundary-determining sets. For example, if e and f are elements of $V_{B,d}$ with e having 2 boundary-determining sets (e_1, e_2) and f having 3 boundary-determining sets (f_1, f_2, f_3), the possible combinations would be: $e_1f_1, e_1f_2, e_1f_3, e_2f_1, e_2f_2, e_2f_3$.

Step 6: Depending on whether either BVA or ECP is used, either a boundary-value set or a nominal-value set is assigned to each possible combination of determining sets. The relationship between a combination of determining sets and a boundary value set is such that if the determining variables e and f take values from a combination $e_p f_q$, then the dependent variable d can only assume a value from the corresponding d_r only.

Step 7: Create all possible combinations of the boundary-determining sets for each boundary-determining variable. For example, if e, f , and g are elements of B with 2, 3, and 2 determining sets respectively, this will result in 12 possible combinations.

$e_1f_1g_1, e_1f_1g_2, e_1f_2g_1, e_1f_2g_2,$
 $e_1f_3g_1, e_1f_3g_2, e_2f_1g_1, e_2f_1g_2,$
 $e_2f_2g_1, e_2f_2g_2, e_2f_3g_1, e_2f_3g_2.$

Step 8: For each combination created in Step 7, assign the boundary-value sets or nominal-value sets for each of the dependent variables according to the associations established in Step 6.

Step 9: Assign values to all the independent variables for each combination, generating all possible sets of independent variables.

Step 10: Conduct traditional BVA or ECP on each set of independent variables to create test cases.

In our study, the technique for combination of input values is one of the following techniques: experience-based, weak, and strong combination.

- Experience-based combination is referred to as the combination technique applied in [10].
- Weak combination (a.k.a. 1-way) chooses one variable value from each partition subset (i.e., equivalence class) such that all classes are covered.
- Strong combination (a.k.a. n-way) is based on the Cartesian product of partition subsets, i.e., testing all interactions of all equivalence classes.

2. NextDate Example

Using the NextDate function as an example, we can demonstrate the outcomes of each step outlined in Section II.1. The NextDate function is a function that takes three variables: Month, Day, and Year, and returns the date of the day after the input date. In this function, the boundary values for Day depends on the values of Month and Year.

Step 1: $D = \{Day\}$, $B = \{Month, Year\}$, $I = \emptyset$.

Step 2: $V_{B,Day} = \{Month, Year\}$.

Step 3: If BVA is used, sets can be created as follows:

$Day_1 = \{1 \dots 31\}$, $Day_2 = \{1 \dots 30\}$,

$Day_3 = \{1 \dots 28\}$, $Day_4 = \{1 \dots 29\}$.

If ECP is used, sets can be created as follows:

$Day_1 = \{15\}$, $Day_2 = \{15\}$,

$Day_3 = \{15\}$, $Day_4 = \{15\}$.

Step 4: If BVA is used, sets can be created as follows:

$Month_1 = \{1, 3, 5, 7, 8, 10, 12\}$ (31 days),

$Month_2 = \{2\}$ (28 or 29 days),

$Month_3 = \{4, 6, 9, 11\}$ (30 days).

$Year_1 = \{1904, 1908, 1912, \dots, 2000\}$ (leap years),

$Year_2 = \{1900, 1901, 1902, 1903, 1905, \dots, 1999\}$.

(non-leap years)

If ECP is used, sets can be created as follows:

$Month_1 = \{7\}$ (31 days),

$Month_2 = \{2\}$ (28 or 29 days),

$Month_3 = \{9\}$ (30 days).

$Year_1 = \{1948\}$ (leap years),

$Year_2 = \{1950\}$ (non-leap years).

Step 5: Create all possible combinations of boundary-determining sets for each dependent variable. In the case of the NextDate function, the only dependent variable is Day. Its determining value sets are as follows:

$Month_1Year_1$, $Month_1Year_2$,

$Month_2Year_1$, $Month_2Year_2$,

$Month_3Year_1$, $Month_3Year_2$.

Step 6: If BVA is used, assign a boundary value set or if ECP is used, assign a nominal-value set to each combination of determining sets. Table I presents an assignment of the value sets of Day to each combination of determining sets $\{Month, Year\}$.

Table I
ASSIGNMENT OF VALUE SETS OF DAY TO
EACH COMBINATION OF DETERMINING SETS.

Determining value sets/Combinations	Boundary-value set/Nominal-value sets
$Month_1Year_1$	Day_1
$Month_1Year_2$	Day_1
$Month_2Year_1$	Day_4
$Month_2Year_2$	Day_3
$Month_3Year_1$	Day_2
$Month_3Year_2$	Day_2

Step 7: The possible combinations for all determining sets are identical to those listed in Step 5.

Step 8: Assign boundary-value sets or nominal-value sets to each combination from Step 7. The following sets are created:

$Month_1Year_1Day_1$, $Month_1Year_2Day_1$,

$Month_2Year_1Day_4$, $Month_2Year_2Day_3$,

$Month_3Year_1Day_2$, $Month_3Year_2Day_2$.

Step 9: This step is omitted due to the absence of independent variables.

Step 10: The results, obtained by applying the testing tool, are presented at the end of Section IV.

III. TESTING TOOL IMPLEMENTATION

Our automated test case generation tool was developed in C# using the .NET framework. Screens of the modules are provided to a user as forms. The generated test cases are displayed in reports. Figure 1 illustrates the process of generating test cases using our new testing tool.

The tool offers six screens to the user, with each screen representing a different module of the testing tool. Table II presents a comparison of the modules between the tool proposed in [10] and our new tool. It lists changes between two tools.

There are eight modules in our tool. These modules are: getVariables, getRelations, getNumOfSetsAndTechnique, getBounds, getNoms, getSetRelationsAndCombinationTechnique, generateBVATestCases and generateECPTTestCases. Each module collects input from the user and forwards its output to the next module in sequence.

In our tool, two modules getNumOfSetsAndTechnique and getSetRelationsAndCombinationTechnique extends two modules getNumOfSets and getSetRelations in the tool proposed in [10], respectively. This extension allows a user to be able to make a choice in applying BVA or ECP and

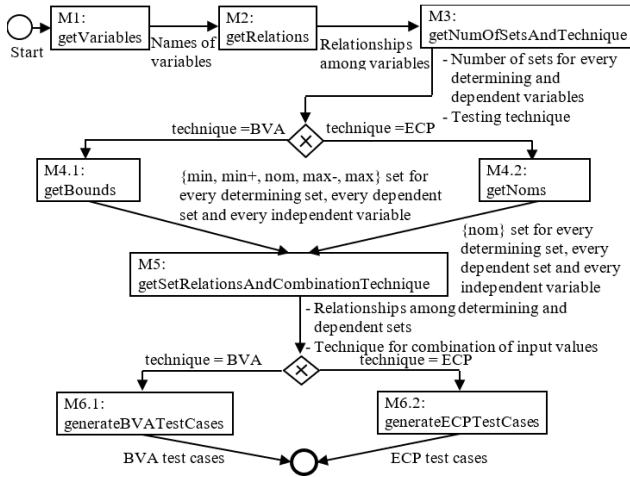


Figure 1. Process of generating test cases using the new testing tool

Table II
COMPARISON OF THE TOOL PROPOSED IN [10] AND THE NEW TOOL.

Module in the tool proposed in [10]	Module in new tool
Module 1: getVariables	Module 1: getVariables (no change)
Module 2: getRelations	Module 2: getRelations (no change)
Module 3: getNumOfSets	Module 3: getNumOfSetsAndTechnique (extended)
Module 4: getBounds	Module 4.1: getBounds (no change) Module 4.2: getNoms (new)
Module 5: getSetRelations	Module 5: getSetRelationsAndCombinationTechnique (extended)
Module 6: generateBVATestCases	Module 6.1: generateBVATestCases (no change) Module 6.2: generateECPTTestCases (new)

a combination technique (experience-based, weak or strong combination). In addition, two new modules getNoms and generateECPTTestCases will be used by the user to collect boundary or nominal values. The design of the modules is briefly described below.

Module 1: getVariables. This module prompts the user to input the names of up to 10 variables.

Module 2: getRelations. This module allows the user to specify the relationships between variables (using checkboxes), enabling the tool to identify boundary-determining, boundary-dependent, and independent variables.

Module 3: getNumOfSetsAndTechnique. This module asks the user to specify the number of sets that can be created for each boundary-dependent, boundary-determining and independent variables. These sets are known as boundary-dependent, boundary-determining and independent sets. Besides, the user is required to choose a black-box testing technique (i.e., BVA or ECP).

Module 4.1: getBounds. This module asks the user to

provide the {min, min+, nom, max-, max} values for each independent variable, as well as for all boundary-dependent and boundary-determining sets.

Module 4.2: getNoms. This module asks the user to enter the {nom} values for each independent variable, and for all boundary-dependent and boundary-determining sets.

Module 5: getSetRelationsAndCombinationTechnique. This module creates all possible combinations of boundary-determining sets for each boundary-dependent variable. It prompts the user to link each combination of boundary-determining sets with a boundary-dependent set. For each combination, the module displays n radio buttons, where n is the total number of sets for the corresponding boundary-dependent variable. In addition, the user is required to choose a combination technique.

Module 6.1: generateBVATestCases and Module 6.2: generateECPTTestCases. These modules create all possible combinations of boundary-determining sets. They then match each combination with a boundary-dependent set for each boundary-dependent variable, according to user-defined associations and the combination technique specified in Module 5. Next, they link all combinations with the independent variables. Finally, Modules 6.1 and 6.2 generate test cases.

IV. DEMONSTRATION OF THE TOOL

We will use the NextDate function to illustrate an application of the tool. We follow the process of generating test cases as introduced in Section III.

Maximum number of variables: 10
Please enter names of variables:

Variable 1	Day
Variable 2	Month
Variable 3	Year
Variable 4	
Variable 5	
Variable 6	
Variable 7	
Variable 8	
Variable 9	
Variable 10	

Figure 2. User Screen 1 - getVariables

Figure 2 displays User Screen 1 (getVariables), where the user enters the variable names such as Day, Month, and Year, then clicks "Submit Query".

Total number of variables: 3
Please enter relations between variables:

Variable		1	2	3
1. Day	depends on	☐	☒	☒
2. Month	depends on	☐	☐	☐
3. Year	depends on	☐	☐	☐

Submit Query

Figure 3. User Screen 2 - getRelations

Figure 3 depicts User Screen 2 (getRelations), where the user defines the relationships between the input variables. For instance, Day is dependent on Month and Year.

Total number of variables: 3
Variables have the following functional dependencies:

{Day} depends on {Month, Year}
{Month} depends on no other variables
{Year} depends on no other variables

Dependent Variables	Number of Sets
Day	4

Determining Variables	Number of sets
Month	3
Year	2

Independent Variables	Number of sets

Testing technique ☒ BVA ☐ ECP

Submit Query

Figure 4. User Screen 3 - getNumOfSetsAndTechnique

Figure 4 illustrates User Screen 3 (getNumOfSetsAndTechnique). This screen prompts the user to input the number of possible sets for each boundary-dependent, boundary-determining and independent variable. There is no independent variable in this example. The testing technique is also chosen by the user.

- Day values can be categorized into four sets:
{Day: $1 \leq \text{Day} \leq 31$ }, {Day: $1 \leq \text{Day} \leq 30$ }, {Day: $1 \leq \text{Day} \leq 28$ } and {Day: $1 \leq \text{Day} \leq 29$ }.
- Month values can be categorized into three sets:
{Month: Month has 31 days}, {Month: Month is February with 28 or 29 days} and {Month: Month has 30 days}.
- Year values can be categorized into two sets:
{Leap years: (Year mod 4 = 0 and Year mod 100 != 0) or (Year mod 400 = 0)} and {Non-leap years: (Year mod 4 != 0) or (Year mod 100 = 0)}.

If the BVA option has been chosen, User Screen 4 (getBounds) is displayed as shown in Figure 5. Here, the boundary values {min, min+, nom, max-, max} for all

Please enter the following bounds:

Independent Variables	min	min+	nom	max-	max+
Day1	1	2	15	30	31
Day2	1	2	15	29	30
Day3	1	2	15	27	28
Day4	1	2	15	28	29

Determining Variables	min	min+	nom	max-	max+
Month1	1	3	7	10	12
Month2	2	2	2	2	2
Month3	4	6	9	9	11
Year1	1904	1908	1948	1996	2000
Year2	1900	1901	1950	1998	1999

Submit Query

Figure 5. User Screen 4 - getBounds

the boundary-dependent and boundary-determining sets are identified. If the ECP option has been chosen, User Screen 5 (getNoms) is displayed as presented in Figure 6. The nominal values {nom} are specified for all the boundary-dependent and boundary-determining sets.

Please enter the following nominal values:

Independent Variables	nom
Day1	15
Day2	15
Day3	15
Day4	15

Determining Variables	nom
Month1	7
Month2	2
Month3	9
Year1	1948
Year2	1950

Submit Query

Figure 6. User Screen 5 - getNoms

Figure 7 presents User Screen 6 (getSetRelationsAndCombinationTechnique), where all possible combinations of the boundary-determining variables are generated. Each combination is linked to a boundary-dependent set. The resulting sets are created for either BVA or ECP.

A boundary value in {determining variable set} generates	====>	A boundary value in (dependent variable set)
For dependent variable {Day} - Total combinations: 6		
{Month ₁ , Year ₁ } ==>	☒ Day ₁	☐ Day ₂ ☐ Day ₃ ☐ Day ₄
{Month ₁ , Year ₂ } ==>	☒ Day ₁	☐ Day ₂ ☐ Day ₃ ☐ Day ₄
{Month ₂ , Year ₁ } ==>	☐ Day ₁ ☐ Day ₂ ☐ Day ₃	☒ Day ₄
{Month ₂ , Year ₂ } ==>	☐ Day ₁ ☐ Day ₂ ☒ Day ₃	☐ Day ₄
{Month ₃ , Year ₁ } ==>	☐ Day ₁ ☒ Day ₂ ☐ Day ₃	☐ Day ₄
{Month ₃ , Year ₂ } ==>	☐ Day ₁ ☒ Day ₂ ☐ Day ₃	☐ Day ₄
Technique of combination	☒ Experience based	☐ Weak ☐ Strong
Submit Query		

Figure 7. User Screen 6 - getSetRelationsAndCombinationTechnique

{Month₁, Year₁, Day₁ }
 {Month₁, Year₂, Day₁ }
 {Month₂, Year₁, Day₄ }
 {Month₂, Year₂, Day₃ }
 {Month₃, Year₁, Day₂ }
 {Month₃, Year₂, Day₂ }

BVA and Experience-based Combination: If the user selects two options BVA and experience-based combination and the boundary values collected as shown in Figure 5, test cases generated as below. These test cases are identical to the ones shown in [10].

- BVA and Experience-based Combination Set #1 {Month₁, Year₁, Day₁}: (1, 1948, 15), (3, 1948, 15), (7, 1948, 15), (10, 1948, 15), (12, 1948, 15), (7, 1904, 15), (7, 1908, 15), (7, 1996, 15), (7, 2000, 15), (7, 1948, 1), (7, 1948, 2), (7, 1948, 30), (7, 1948, 31).
- BVA and Experience-based Combination Set #2 {Month₁, Year₂, Day₁}: (1, 1950, 15), (3, 1950, 15), (7, 1950, 15), (10, 1950, 15), (12, 1950, 15), (7, 1900, 15), (7, 1901, 15), (7, 1998, 15), (7, 1999, 15), (7, 1950, 1), (7, 1950, 2), (7, 1950, 30), (7, 1950, 31).
- BVA and Experience-based Combination Set #3 {Month₂, Year₁, Day₄}: (2, 1948, 15), (2, 1948, 15), (2, 1948, 15), (2, 1948, 15), (2, 1904, 15), (2, 1908, 15), (2, 1996, 15), (2, 2000, 15), (2, 1948, 1), (2, 1948, 2), (2, 1948, 28), (2, 1948, 29).
- BVA and Experience-based Combination Set #4 {Month₂, Year₂, Day₃}: (2, 1950, 15), (2, 1950, 15), (2, 1950, 15), (2, 1950, 15), (2, 1900, 15), (2, 1901, 15), (2, 1998, 15), (2, 1999, 15), (2, 1950, 1), (2, 1950, 2), (2, 1950, 27), (2, 1950, 28).
- BVA and Experience-based Combination Set #5 {Month₃, Year₁, Day₂}: (4, 1948, 15), (6, 1948, 15), (9, 1948, 15), (9, 1948, 15), (11, 1948, 15), (9, 1904, 15), (9, 1908, 15), (9, 1996, 15), (9, 2000, 15), (9,

1948, 1), (9, 1948, 2), (9, 1948, 29), (9, 1948, 30).

- BVA and Experience-based Combination Set #6 {Month₃, Year₂, Day₂}: (4, 1950, 15), (6, 1950, 15), (9, 1950, 15), (9, 1950, 15), (11, 1950, 15), (9, 1900, 15), (9, 1901, 15), (9, 1998, 15), (9, 1999, 15), (9, 1950, 1), (9, 1950, 2), (9, 1950, 29), (9, 1950, 30).

BVA and Weak Combination: If the user selects BVA and weak combination options and enters the boundary values as shown in Figure 5, test cases generated as below.

- BVA and Weak Combination Set #1 {Month₁, Year₁, Day₁}: (1, 1904, 1), (3, 1908, 2), (7, 1948, 15), (10, 1996, 30), (12, 2000, 31).
- BVA and Weak Combination Set #2 {Month₁, Year₂, Day₁}: (1, 1900, 1), (3, 1901, 2), (7, 1950, 15), (10, 1998, 30), (12, 1999, 31).
- BVA and Weak Combination Set #3 {Month₂, Year₁, Day₄}: (2, 1904, 1), (2, 1908, 2), (2, 1948, 15), (2, 1996, 28), (2, 2000, 28).
- BVA and Weak Combination Set #4 {Month₂, Year₂, Day₃}: (2, 1900, 1), (2, 1901, 2), (2, 1950, 15), (2, 1998, 27), (2, 1999, 28).
- BVA and Weak Combination Set #5 {Month₃, Year₁, Day₂}: (4, 1904, 1), (6, 1908, 2), (9, 1948, 15), (9, 1996, 29), (11, 2000, 30).
- BVA and Weak Combination Set #6 {Month₃, Year₂, Day₂}: (4, 1900, 1), (6, 1901, 2), (9, 1950, 1), (9, 1998, 29), (11, 1999, 30).

BVA and Strong Combination: If BVA and strong combination options are selected and the boundary values are collected as shown in Figure 5, test cases generated as below.

- BVA and Strong Combination Set #1 {Month₁, Year₁, Day₁}: (1, 1904, 1), (1, 1904, 2), (1, 1904, 15), (1, 1904, 30), (1, 1904, 31), (1, 1908, 1), (1, 1908, 2), (1, 1908, 15), (1, 1908, 30), (1, 1908, 31), (1, 1948, 1), (1, 1948, 2), (1, 1948, 15), (1, 1948, 30), (1, 1948, 31), (1, 1996, 1), (1, 1996, 2), (1, 1996, 15), (1, 1996, 30), (1, 1996, 31), (1, 2000, 1), (1, 2000, 2), (1, 2000, 15), (1, 2000, 30), (1, 2000, 31), (3, 1904, 1), (3, 1904, 2), (3, 1904, 15), (3, 1904, 30), (3, 1904, 31), (3, 1908, 1), (3, 1908, 2), (3, 1908, 15), (3, 1908, 30), (3, 1908, 31), (3, 1948, 1), (3, 1948, 2), (3, 1948, 15), (3, 1948, 30), (3, 1948, 31), (3, 1996, 1), (3, 1996, 2), (3, 1996, 15), (3, 1996, 30), (3, 1996, 31), (3, 2000, 1), (3, 2000, 2), (3, 2000, 15), (3, 2000, 30), (3, 2000, 31), (7, 1904, 1), (7, 1904, 2), (7, 1904, 15), (7, 1904, 30), (7, 1904, 31), (7, 1908, 1), (7, 1908, 2), (7, 1908, 15), (7, 1908, 30), (7, 1908, 31), (7, 1948, 1), (7, 1948, 2), (7, 1948, 15), (7, 1948, 30), (7, 1948, 31), (7, 1996, 1), (7, 1996, 2), (7, 1996, 15), (7, 1996, 30), (7, 1996, 31), (7, 2000, 1), (7, 2000, 2), (7, 2000, 15), (7, 2000, 30), (7, 2000,

31), (10, 1904, 1), (10, 1904, 2), (10, 1904, 15), (10, 1904, 30), (10, 1904, 31), (10, 1908, 1), (10, 1908, 2), (10, 1908, 15), (10, 1908, 30), (10, 1908, 31), (10, 1948, 1), (10, 1948, 2), (10, 1948, 15), (10, 1948, 30), (10, 1948, 31), (10, 1996, 1), (10, 1996, 2), (10, 1996, 15), (10, 1996, 30), (10, 1996, 31), (10, 2000, 1), (10, 2000, 2), (10, 2000, 15), (10, 2000, 30), (10, 2000, 31), (12, 1904, 1), (12, 1904, 2), (12, 1904, 15), (12, 1904, 30), (12, 1904, 31), (12, 1908, 1), (12, 1908, 2), (12, 1908, 15), (12, 1908, 30), (12, 1908, 31), (12, 1948, 1), (12, 1948, 2), (12, 1948, 15), (12, 1948, 30), (12, 1948, 31), (12, 1996, 1), (12, 1996, 2), (12, 1996, 15), (12, 1996, 30), (12, 1996, 31), (12, 2000, 1), (12, 2000, 2), (12, 2000, 15), (12, 2000, 30), (12, 2000, 31).

- Similarly, BVA and Strong Combination Set #2 $\{Month_1, Year_2, Day_1\}$, BVA and Strong Combination Set #3 $\{Month_2, Year_1, Day_4\}$, BVA and Strong Combination Set #4 $\{Month_2, Year_2, Day_3\}$, BVA and Strong Combination Set #5 $\{Month_3, Year_1, Day_2\}$, BVA and Strong Combination Set #6 $\{Month_3, Year_2, Day_2\}$ are created using the Cartesian product of the corresponding sets.

ECP and Weak Combination: If the user selects ECP and weak combination options and enters the nominal values as shown in Figure 6, test cases generated as below:

- ECP and Weak Combination Set #1 $\{Month_1, Year_1, Day_1\}$: (7, 1948, 15).
- ECP and Weak Combination Set #2 $\{Month_1, Year_2, Day_1\}$: (7, 1950, 15).
- ECP and Weak Combination Set #3 $\{Month_2, Year_1, Day_4\}$: (2, 1948, 15).
- ECP and Weak Combination Set #4 $\{Month_2, Year_2, Day_3\}$: (2, 1950, 15).
- ECP and Weak Combination Set #5 $\{Month_3, Year_1, Day_2\}$: (9, 1948, 15).
- ECP and Weak Combination Set #6 $\{Month_3, Year_2, Day_2\}$: (9, 1950, 15).

The process of generating test cases using the tool offers a variety of paths designed to provide flexibility in the choice of testing and combination techniques. Adaptive design allows the tool to adjust its behavior, process, and interface according to the user's needs. Exhaustive testing is not possible. In addition, each test case adds to the cost of testing. Therefore, the user may need a suitable technique to choose a subset of all the combinations.

The ECP technique allows the user to need less effort to collect input values and reduce the number of test cases to be executed due to just selecting one representative value from each equivalence class (subset) as test data. When the ECP and Weak Combination option is chosen, a minimal

set of six test cases is created for six possible combinations of the boundary-determining variables. One variable value is selected from each equivalence class (one Day_i , $Month_i$, and $Year_i$) such that all classes are covered. However, the ECP option ignores testing boundary values.

On the other hand, the BVA technique focuses on testing the boundary or edge values of input ranges and thus needs a greater number of test cases. To test every boundary value at least once, we need five test cases {min, min+, nom, max-, max} from each equivalence class. When the BVA and Weak Combination option is chosen, a set of thirty test cases is created for six possible combinations of the boundary-determining variables. These test cases are good enough to cover all equivalence classes and all boundary values. Therefore, the BVA and Weak Combination option may be more cost-effective than the BVA and Experience-based Combination option (used in the original algorithm) and BVA and Strong Combination option.

V. CONCLUSION

This paper addresses some limitations on the testing case design algorithm using a divide-and-rule approach in terms of advancement to increase the flexibility of the algorithm. The core idea is that the extended algorithm considers the integrated use of BVA, ECP and different combination techniques. Applying the extended algorithm described in this paper allows a user more options in selecting input values and output test data sets. This may reduce the cost of testing. The tool for automated test case generation was developed in C# language and can be effortlessly utilized by navigating the user interface and supplying the required inputs. Currently, the extended algorithm supports test case generation from numerical values. As a future direction, we plan to employ this algorithm to verify a mixing of different data types such as Boolean, numeric and string variables.

ACKNOWLEDGEMENT

The authors would like to thank the anonymous reviewers for their comments on an earlier version of this paper. The authors acknowledge Can Tho University for financial support of this work. This study is funded in part by the Can Tho University, Code: T2023-123.

REFERENCES

- [1] Pressman R. and Maxim B. "Software Engineering - A Practitioner's approach.", 9th Edition. The McGraw Hill Companies, Inc., New York (2020).
- [2] Rao C. et al. "Combinatorial Test Generation for Multiple Input Models With Shared Parameters.", *IEEE Transactions on Software Engineering*, vol. 48, no. 7, pp. 2606-2628 (2022).
- [3] Nie C. and Leung H. "A survey of combinatorial testing.", *ACM Comput. Surv.* 43, 2, Article 11 (2011).

- [4] Chen T.Y. and Poon P.L., Tang S.F., and Tse T.H. "On the identification of categories and choices for categorypartition test case generation.", *Information and Software Technology*, 46 (13) pp. 887-898 (2004).
- [5] Chen T.Y., Cheng M.Y., Poon P.L., Tse T.H., and Yu Y.T. "A study of input domain partitioning.", *Proceedings of the 20th IASTED International Multi-Conference on Applied Informatics (AI 2002)*, ACTA Press, Calgary, Canada, pp. 176-181 (2002).
- [6] Lei Y., Kacker R., Kuhn D. R., Okun V., and Lawrence J. "IPOG: A General Strategy for T-Way Software Testing.", *14th Annual IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'07)*, Tucson, AZ, USA, pp. 549-556 (2007).
- [7] Debnath N. and Haakenson J.R. "Automated testing design and description for certain functions.", *Proceedings of the Sixth IEEE International Conference on Electro/Information Technology, (IEEE-EIT 2006)*, Michigan State University, East Lansing, MI, USA (2006).
- [8] Feng W. and Zhang Z. "Sequence algorithms for boundary value analysis with constrained input parameters.", *Proceedings of the ISCA 14th International Conference on Intelligent and Adaptive Systems and Software Engineering (IASSE-2005)*, Toronto, Canada, pp. 255-260 (2005).
- [9] Burnstein, I. "Practical Software Testing: A Process-Oriented Approach.", *Springer*, New York (2006).
- [10] Vij K. and Feng W. "Boundary Value Analysis Using Divide-and-Rule Approach.", *Fifth International Conference on Information Technology: New Generations (itng 2008)*, Las Vegas, NV, USA, pp. 70-75 (2008).



Danh Nguyen-Cong received his B.S degree in Computer Science from Can Tho University in 2000. In 2006, he received his M.S. degree in Information Technology from King Mongkut's University of Technology North Bangkok, Thailand. In 2009, he worked as a researcher at the Software Engineering Research Group (AGSE), the

Technical University of Kaiserslautern, Germany. He completed his PhD in Computer Science at CNRS, LIMOS UMR 6158, Clermont Auvergne University, France in 2018. His research interests include Big Data, Software Process, and Software Quality Assurance and Testing.

Email: ncdanh@cit.ctu.edu.vn

Utilizing Blockchain Technology to Secure Database from Unauthorized Modifications

Cu Vinh Loc, Tran Thi Dinh Xuyen, Truong Xuan Viet, Tran Hoang Viet, Le Hoang Thao, Truong Minh Thai
Can Tho University, Can Tho, Vietnam

Correspondence: Cu Vinh Loc, cvloc@ctu.edu.vn

Received: 14/07/2024, revised: 03/10/2024, accepted: 09/10/2024

Digital Object Identifier: 10.32913/mic-ict-research-vn.v2024.n3.1330

Abstract: Centralized database-based applications are frequently susceptible to internal threats. The data stored in a relational database can be changed by any user who has administrative access to the hosting server or database system. Moreover, this kind of user may alter the associated log files, which would make it very challenging to identify any illegal alteration. It would also be difficult to attribute the assault to privileged users as well. In this study, we propose a solution to build a service on the context of student score management based on Blockchain technology. A database system based on the Blockchain platform along with accompanying services is built to allow storage and query of student learning outcomes. The database security system operates in parallel with the current system while still ensuring transparency, security and integrity of data. We develop a verification module that uses Blockchain to verify the integrity of the corresponding tuples and identify any unwanted modifications. Oracle database, RESTful API, and Hyperledger Fabric have all been used in the implementation of our method. Our findings show that the overhead of integrity checking remains constant per tuple in the results of a query and increases linearly using real data and SQL queries of different types and complexity.

Keywords: *Blockchain, hyperledger, centralized database systems tamper resistance features.*

Tiêu đề: Giám sát nhiệt độ kho lạnh bằng phương pháp nội suy ba chiều

Tóm tắt: Internet vạn vật (IoT) được ứng dụng rộng rãi trong nông nghiệp và công nghiệp. Bài báo này đề xuất một mô hình giám sát nhiệt độ ba chiều cho kho lạnh, tăng cường khả năng quan sát và phát hiện những thay đổi nhiệt độ bất thường trong kho. Sử dụng việc thu thập nhiệt độ theo thời gian thực và áp dụng các kỹ thuật nội suy tam tuyến, mô hình này xác định nhiệt độ tại tất cả các vị trí trong kho. Nó chấp nhận các giá trị nhiệt độ và vị trí cảm biến, phân đoạn không gian kho lạnh một cách thích hợp để áp dụng nội suy ba tuyến tính và xác định giá trị nhiệt độ tại mọi vị trí trong kho. Mô hình đã được thử nghiệm dựa trên số lượng cảm biến khác nhau và các vị trí cảm biến khác nhau. Dựa trên đó, chúng tôi cung cấp một giải pháp tối ưu cho số lượng cảm biến cần thiết cho kho lạnh.

Từ khóa: *Blockchain, hyperledger, centralized database systems tamper resistance features.*

I. INTRODUCTION

Enhancing the quality of management and training of educational and training institutions is one of the important goals for developing an effective and sustainable human resource. However, the current education and training system still has certain limitations, the security and storage of centralized data are increasingly improved, but the disadvantages have not been overcome. Private student records hold significant value for both individuals and universities alike. The existing training management system can guarantee that operational faults are kept to a minimum or that outside agent attacks are avoided. This is not the best course of action, though, as there are still issues with making sure that data is accurate during management and with

detecting inaccurate or erroneous data. In particular, data can still be illegally or incorrectly altered during operation, and the current system has trouble identifying incidents. Certain vulnerabilities in the integrity and security of the current system have been made clear by the previously described considerations. Data is also not carefully controlled, and identifying and correcting problems in it requires a significant amount of time and effort. Specifically, there is a lack of transparency for fast fixes due to the way learning outcomes are maintained and kept, which makes it challenging to check and identify problems.

A malicious attack on a computer system or network carried out by an individual with authorized system access is known as an inside attack. Since insiders have authorized

access and may be knowledgeable about system policies and procedures as well as network architecture, they have a clear advantage over external attackers. Additionally, because many firms prioritize safeguarding against outsiders, there can be less security against insider attacks. Insider threats are those who intentionally or unintentionally create vulnerabilities or who use permitted access to an organization's cyberassets for malevolent or unauthorized reasons. They could be contracted workers, direct employees, or outside providers of computing and data services. Insider threat cases account for around 23% of all cybercrime occurrences, a ratio that has remained constant over the years despite a large increase in the overall number of attacks. Over 50% of firms report having experienced an insider cyberattack annually. The studies also reveal that insider assaults continue to be the hardest to identify, with 90% of firms believing they are vulnerable to insider threats.

In a relational database system, conventional integrity constraints refer to the process of guaranteeing the integrity of tuples in accordance with established constraints, such as data types of attribute values, foreign key constraints, etc. Malicious alteration of the tuples in a database management system (DBMS) is another source of integrity issues. Firewalls and access control measures are typically used to protect this data integrity. A DBMS's access control policies limit the number of users who are granted administrative privileges. The DBMS server cannot be directly accessed by an outsider thanks to firewalls. On the other hand, an internal modification occurs when a user with legal rights, such as an administrator, abuses their position to get or alter data. Internal modifications are now considered a serious and non-trivial threat to a database management system's ability to protect data integrity.

Passive or active internal alterations are both possible. While active modifications entail manipulating the data and logs to change query results, passive modifications involve obtaining and using the data in an unethical manner. Academic grade record tampering is a straightforward yet insightful example of the second form of attack, and it has been documented on several occasions in the recent past. In order to eliminate paper ballots from elections, several nations have embraced electronic voting machines. Every vote cast on an electronic voting machine is treated as a transaction by the embedded DBMS. Additionally susceptible to tampering and insider attacks are these computerized voting equipment. In terms of effective, varied, and scalable data storage options, traditional databases have advanced significantly over the previous few decades. The application has been effectively addressed by SQL query processing and optimization in conjunction with contemporary hardware. However, as previously mentioned, the modern issues

include data security and integrity, of which detecting and preventing internal modification are essential components.

Blockchain is a new technology that offers high assurances of tamper resistance and immutability for decentralized data storage. Blockchains are comparable to native database management systems (DBMSs) that only append logs. But in an internal modification, the illegal users can change login records and logs to erase evidence of data manipulation because they have administrative access. Placing all of the databases on blockchain frameworks would be an easy fix. The majority of Blockchain frameworks and DBMSs enabled by Blockchain technology lack the robust SQL querying interface found in most contemporary DBMSs. Concern over data privacy is also growing as more and more data is being uploaded to public Blockchain networks.

We have found Blockchain being applied to solve issues in many fields like educational data security [1, 2], student certificate management [3], relational database security [4], and traceability management [2, 5–8]. Similar to the existing works, our approach allows storing student grades with authorization. The data will be checked and validated in off-chain database (Oracle) and on-chain database (Blockchain). Smart contract is used to maintain security and consistency. Figure 1 depicts the workflow of the proposed system.

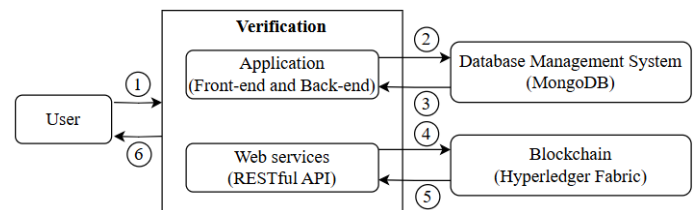


Figure 1. An overview diagram of the proposed approach.

The paper makes the following primary contributions.

- We develop Blockchain database, storage, and query services. The solution makes it feasible to validate and save course scores..
- A history of the creation and adjustment of course scores will be stored on the Blockchain throughout time.
- The query speed of Blockchain and the security of traditional databases are overcome when traditional data storage (including relational and non-relational) and Blockchain are combined.
- Consensus procedures along with the limitations and logic found in smart contracts guarantee that data is reliable and consistent at all times.

II. RELATED WORK

Ali *et al.* [9] put forward three strategies for deploying Blockchain technology to create a fully operational student information system that preserves transactions pertaining to student marks, course registration data, and the records of both students and instructors. Moreover, staying away from positions where data integrity is at risk, such as super administrator or centrally exposed storage. Employing the suggested models facilitates the transition to an electronic community in which legitimate certifications may be quickly generated and disseminated to interested parties without requiring the involvement of a centralized administration. In the work [1], the authors investigated the use of Ethereum's Proof of Authority (PoA) consensus method to create an agent-based Blockchain model system for protecting educational data against insider threats. The study established a basis for furthering the present comprehension of Blockchain systems, creating Blockchain models utilizing the technology's unchangeable characteristics, and integrating intelligent agents. Liang and Xu [10] have designed and implemented a decentralized system for managing student score sharing using blockchain technology. Furthermore, they guarantee the traceability, tamper-proof, distributed storage, and security of the query process by utilizing the Ethereum smart contract to control on-chain and off-chain data storage in addition to on-chain inquiry and verification. The study concludes with examining the development of a prototype system based on the Ethereum framework. The outcomes of the experiment demonstrate the viability of the anti-tampering and traceability plan put forth in this work. Blockchain technology can completely meet the needs of different stakeholders involved in student performance management. Student score information is completely protected because each subject's data is encrypted using a unique private key.

With the introduction of the work [4], the cloud relational database (RDB) known as BC over cloud-RDB received an improved structure based on blockchain technology (BC). It allows the client to identify and stop incorrect RDB manipulation through a self-verification method. Agile BC-based RDB and secure BC-based RDB are the two systems that the authors proposed to enhance cloud-RDB. Based on the Byzantine Fault Tolerance consensus, both are divided among a number of cloud service providers. Furthermore, both depend on employing the SHA-256 to connect records to one another. In addition, a proof-of-work consensus is used by secure BC-based RDB to prevent data offensive operations. Based on security analysis and system performance, the agile BC-based RDB is strongly recommended for the high throughput database. In order to preserve control and ownership of the credentials, a framework

for issuing student certifications locally as well as online has been proposed by Ghani *et al.* [3]. With the use of blockchain technology, this framework facilitates electronic certification sharing and verification. The implementation of the blockchain-based certification system in universities is expected to result in reduced latency for the purposes of certification issuance, sharing, and verification. In the study, a blockchain-based framework for exchanging e-certifications is suggested. The framework is evaluated by calculating the latency time between transactions and the average time it takes to issue a certificate.

In another work [11], the authors introduce an Ethereum smart contract-based safe datastore. Initially, the authors investigate the degree to which a datastore powered by smart contracts ought to mimic a conventional database system. Second, they look into the best way to minimize gas consumption and maximize flexibility while storing the data in a datastore that is based on smart contracts. Third, they look for clarification on whether sophisticated processing like data encryption and data analytic algorithms should be included in a datastore built on smart contracts. The authors also come to the conclusion that because of the cost and security risks, sophisticated processing shouldn't be permitted in smart contracts. Furthermore, traceability data is also managed using blockchain. Cao *et al.* [7] examine how blockchain technology might support sustainable food supply chains, which adds to the growing body of study in this area. This study suggests a blockchain-enabled architectural framework for reliable communication about the sustainability features of food goods in an effort to promote the use of blockchain for sustainability in the context of the food supply chain. The multi-layered architecture for using blockchain-based traceability to communicate to customers the reliable sustainability features of a specific food product is outlined in the blockchain-enabled architectural framework. This study provides useful information for industry 4.0 researchers and practitioners, bolstering sustainability communication with blockchain-based traceability.

III. METHODOLOGY

The overview of the proposed system can be described as follows. Users will interact with the system via the website. Users will be provided with all functions depending on their role in the system. The system is built with the following components: NodeJS Server, Oracle Database, Hyperledger Fabric blockchain and client using ReactJS. The system uses ReactJS from the web browser using RESTful API (GET, POST, PUT, DELETE) to interact with the server via HTTP method. For the function of saving mark data. Student test marks will be saved simultaneously in the

Oracle database and the blockchain database. When data is written to the blockchain database, the system will not interact directly with the database, instead Hyperledger Fabric designs an interaction model through smart contracts (also known as chaincode) and the communication will be controlled. The blockchain database will store information including student information and academic results, student information. All other data will be stored in the Oracle database. While operating the system, Oracle database will be prioritized because it will have higher retrieval speed, for the use case blockchain database is only used when creating new users, storing student marks.

1. System architecture

The system architecture is as depicted in the Figure 2, where the Hyperledger Fabric network is configured, launched and managed by Docker, “Blockchain Operator” acts as the identity manager and Hyperledger fabric network. The user can interact with the application through ReactJS. The system will interact with the server using NodeJS. To interact with the blockchain network, the software development kit (SDK) will be used for the above purpose.

Figure 2 is an overview of the architectural system for a Hyperledger Fabric network that includes a blockchain operator, Docker, Fabric SDK, ReactJS, and users. This architecture outlines how these components interact with each other in a decentralized application. The blockchain operator manages the Hyperledger Fabric network, including deploying and maintaining the network components like peers, orderers, and certificate authorities (CAs). The blockchain operator’s responsibilities are to configure and manage network settings, monitor the network’s health and performance, manage chaincode deployment and updates, and ensure security and compliance within the network. Docker is used to containerize the various components of the Hyperledger Fabric network, enabling easy deployment, scaling, and management. Docker responsibilities consist of creating isolated environments for peers, orderers, and CAs, ensuring that dependencies and versions do not interfere with one another, facilitating the rapid deployment of network components using Docker Compose, and enabling the running of multiple instances of the blockchain network for development, testing, and production scenarios. The Fabric SDK provides developers with the tools to interact programmatically with the Hyperledger Fabric network. It is responsible for facilitating application development by providing APIs for common tasks (e.g., querying the ledger, submitting transactions), handling user authentication, endorsement, and transaction lifecycle, and supporting

integration with client applications written in various programming languages.

ReactJS serves as the frontend framework for building user interfaces that interact with the Hyperledger Fabric network. Its responsibilities include providing a dynamic and responsive user experience for end-users, facilitating communication with the backend services through API calls, displaying the results of transactions, queries, and historical data in a user-friendly manner, and handling user authentication and session management. Users are the individuals or systems that interact with the application built on top of the Hyperledger Fabric network. Users are responsible for performing actions such as submitting transactions, querying data, and viewing historical records, participating in the blockchain ecosystem by managing their credentials and interacting with the smart contracts (chaincode), and utilizing the application’s interface (built with ReactJS) to engage with the blockchain.

When users interact with the system using chaincode (smart contracts), data is queried or modified within a component that Hyperledger Fabric (or any other blockchain platform) refers to as a ledger. As its name suggests, the ledger maintains a record of all data changes from the system’s inception to the present. In other words, the ledger has both the current value of the properties of the objects and the history of changes that led to this current value. The data will not be stored in the traditional way but will be encrypted and then stored in blocks in the ledger.

Figure 3 shows the architecture of a ledger stored on a peer of the system. A ledger in Hyperledger Fabric consists of two distinct parts like the world state and the blockchain. Each component represents a set of facts about a set of stored objects. First, the world state – a database that holds the current values of a set of ledger states. The world state makes it easy for a program to directly access the current value of a state instead of having to calculate it by going through the entire transaction log. By default, ledger states are represented as key-value pairs and we will see later how Hyperledger Fabric provides flexibility in this regard. The world state can change frequently as states can be created, updated and deleted. Second, the blockchain – which contains a transaction log that records all the changes that led to the current state of the world. The transactions are collected inside blocks that are attached to the blockchain. This allows for a historical understanding of the changes that led to the current state of the world. The blockchain data structure is very different from the current state of the world because once it is written, it cannot be modified because it is immutable. In short, once the data is stored in the blockchain, only the world state in the ledger changes, all data stored in the blockchain cannot be changed

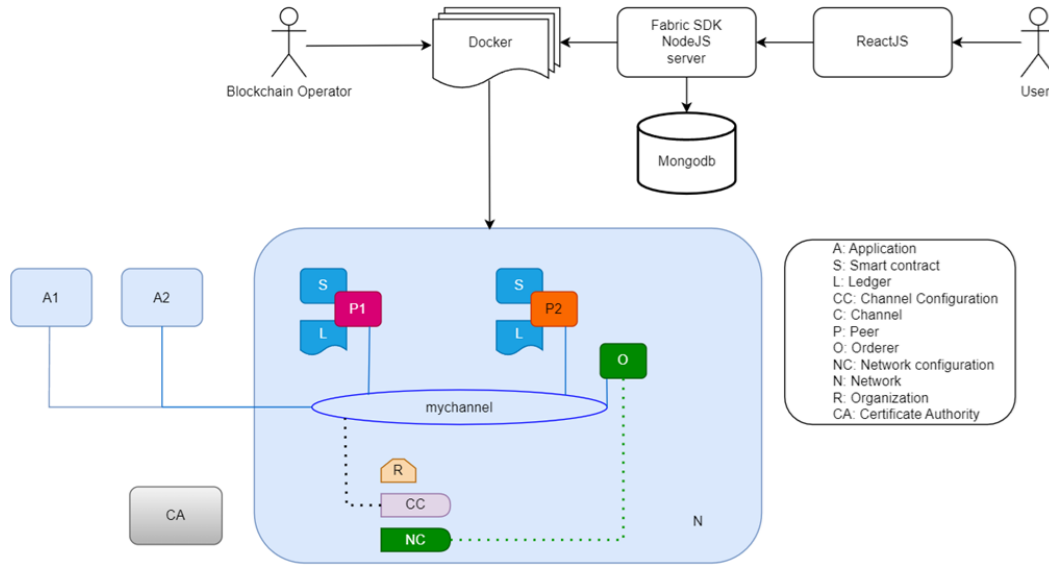


Figure 2. The architecture of the proposed system.

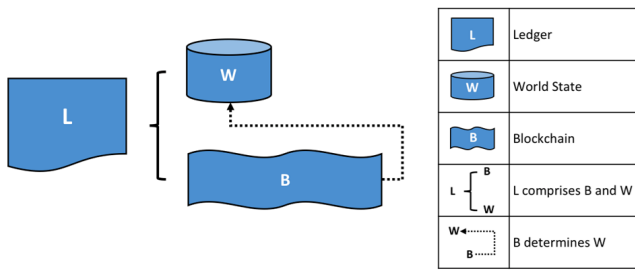


Figure 3. Architecture for hyperledger fabric storage (Source: hyperledger-fabric.readthedocs.io).

but can only be updated.

The system interaction flow can be describes as follows.

- 1) User Interface: Users interact with the application through a web-based interface created with ReactJS. They can log in, submit transactions, and query data.
- 2) ReactJS to Fabric SDK: When a user performs an action (like submitting a transaction), the ReactJS app sends the request to the backend, which utilizes the Fabric SDK to communicate with the Fabric network.
- 3) Fabric SDK to Blockchain Network: The Fabric SDK handles the logic to create transaction proposals, send them to the endorsing peers, collect endorsements, and submit the transactions to the orderer.
- 4) Docker Management: The Ledgers, peers, orderers, and other components of the Fabric network run in Docker containers. The blockchain operator manages these containers using Docker and Docker Compose, facilitating easy orchestration.
- 5) Orderer and Peers Processing: The ordering service

collates transactions and broadcasts them to all peers. Peers validate the transactions and commit them to their local ledgers.

- 6) Data Retrieval: Users can query the blockchain to retrieve historical data. ReactJS forward the requests to the Fabric SDK, which fetches the requested data from the ledger and sends it back to the user interface for display.

Algorithm 1 Data Processing Flow in Hyperledger Network

- 1: Initialize Network Components
- 2: Define Chaincode
- 3: Deploy Chaincode
- 4: Prepare Client Application
- 5: Client Prepares Transaction Proposal
- 6: Send Proposal to Endorsing Peers
- 7: **if** Endorsements are valid **then**
- 8: Submit Endorsed Transaction to Orderer
- 9: Orderer Distributes Block to Peers
- 10: Peers Validate Transactions
- 11: Commit Transaction to Ledger and Update State
- 12: Client Queries Updated World State
- 13: Implement Logging and Monitoring
- 14: **else**
- 15: Handle Invalid Endorsements
- 16: **end if**

2. Data Processing Flow in Hyperledger Fabric

To query or perform functions, the system uses the SDK to query and update the ledger. The steps are as follows.

- Send transaction: Client sends transaction information including sender information to Hyperledger network. SDK takes sender and transaction information to package and propose transaction. After packaging, transaction will be sent to peers in the system for execution.
- Verify signature and execute transaction: The system checks the transaction information for validity before executing the transaction. If the information is invalid, an error will be reported. After being validated, peers will call the chaincode and use the input data to process the data. After executing the transaction, the chaincode will generate the transaction result. The value of the transaction just created will be sent back to the SDK as a “proposal response”.
- SDK compares proposal responses: After receiving the return value from the peer, the system will check the validity of the return value (identity of the peer, return result). If the return transaction is valid, the system checks whether the transaction created is a query or valid. The query transaction will send the result to the client and end the execution process, otherwise it will send the transaction to the ordering service.
- Collect transactions and save them to ledger: The SDK will send data to the ordering service (including transaction proposal and transaction response). The data sent will be in the form of a “Transaction Message”, which contains key-value pairs for the object to be updated. The Ordering Service will arrange the sent transactions in chronological order.
- Authenticate transaction: Transactions after successful arrangement will send updated data to peers. Each transaction will be then tagged as valid or invalid.
- Update ledger: Each peer will add transactions to the block when the transaction is valid. For each valid transaction, data will be written to the ledger. The updated data will be sent to the system so that users can know.

Algorithm 1 is a high-level algorithm that outlines the general steps involved in data processing within a Hyperledger network, particularly focusing on Hyperledger Fabric.

3. Function to save marks to blockchain

The following requirements are met by the terms of the smart contract.

- (1) Check transaction: Verify the mark’s existence in the blockchain, as well as the student’s academic performance. If the performance is already recorded in the blockchain, report a mistake.
- (2) Check user identity: Verify whether the certificate authority has registered the user identity or not. The

system will alert the user to a mistake if they are not in the roles of administrator, instructor, or rector.

- (3) Check the right to update marks: The system will determine whether or not the user still has the authority to save or amend marks if they are a teacher. The system will alert users to an issue if the time for updating the mark has passed.
- (4) Check semester results: Verify whether or not student learning outcomes have been generated for the semester; if they have, the system will display an error.
- The following will be the description of the conditions of the smart contract.

Algorithm 2 Save Student Marks to Blockchain

Require: StudentID, Subject, Marks

Ensure: Transaction is recorded on the blockchain

- 1: **Initialize** Blockchain Network Components
 - 2: **Define** Chaincode for Student Marks
 - 3: **Deploy** Chaincode on the Blockchain Network
 - 4: **Prepare** Client Application with User Credentials
 - 5: **Create** a Transaction Proposal
 - 6: **Set** Proposal Inputs: (StudentID, Subject, Marks)
 - 7: **Send** Proposal to Endorsing Peers
 - 8: **Receive** Endorsements from Peers
 - 9: **if** Endorsements are valid **then**
 - 10: **Submit** the Endorsed Transaction to the Orderer
 - 11: **Wait** for the Orderer to create a Block
 - 12: **Receive** Block from Orderer
 - 13: **Distribute** Block to all Peers
 - 14: **Commit** Transaction to Ledger
 - 15: **Update** World State with Student Marks
 - 16: **Log** Transaction for Auditing
 - 17: **else**
 - 18: **Handle** Invalid Endorsements
 - 19: **Notify** User of Failure
 - 20: **end if**
-

The algorithm 2 outlines the steps needed to create a transaction that records student marks on a blockchain, such as Hyperledger Fabric. The algorithm includes steps for preparing the data, creating a transaction proposal, endorsing it, and committing it to the blockchain.

4. Function to update student marks

The smart contract conditions for updating student marks are described below.

- (1) Check user identity: Verify whether the certificate authority has registered the user identity or not. The system will alert the user to a mistake if they are not in the roles of administrator, instructor, or rector.

- (2) Check the right to update student marks: The system will determine whether or not the user still has the authority to save or amend marks if they are a teacher. The system will alert users to an issue if the time for updating the student mark has passed.
- (3) Check semester results: Verify the semester's learning outcomes; if there are no student learning outcomes, the system will flag an error.
- The conditions of the smart contract are described as follows.

Algorithm 3 Update Student Marks on Blockchain

Require: StudentID, Subject, NewMarks

Ensure: Updated Transaction is recorded on the blockchain

```

1: Initialize Blockchain Network Components
2: Prepare Client Application with User Credentials
3: Retrieve Current Marks from Blockchain using Get-Mark function
4: if Current Marks exist then
5:   Create an Update Transaction Proposal
6:   Set Proposal Inputs: (StudentID, Subject, NewMarks)
7:   Send Proposal to Endorsing Peers for Validation
8:   Receive Endorsements from Peers
9:   if Endorsements are valid then
10:    Submit the Endorsed Update Transaction to the Orderer
11:    Wait for the Orderer to create a Block
12:    Receive Block from Orderer
13:    Distribute Block to all Peers
14:    Commit Updated Transaction to Ledger
15:    Update World State with NewMarks
16:    Log Update Transaction for Auditing
17:  else
18:    Handle Invalid Endorsements
19:    Notify User of Failure
20:  end if
21: else
22:   Notify User that No Current Marks exist for Update
23: end if

```

The algorithm 3 provides a comprehensive approach to updating student marks on a blockchain, ensuring that all necessary checks and validations are performed to maintain the integrity and security of the data.

5. Function to query the history of mark update

The following is a description of the smart contract conditions used to query the mark updating history.

- (1) Check user identity: Verify whether the certificate authority has the user identity registered. The system will raise an error if the user does not have the role of admin or rector, or does not have access to the smart contract.
- (2) Check learning outcome data.
- The following is a description of the smart contract's requirements for retrieving the mark update history

The relationships between the various verification components for every given SQL query are depicted in Figure 1. The SQL processor parses a SQL query that was started via the web application interface (①) and sends it to the database management system (DBMS) with the deliberate changes needed to ensure integrity (②). Tuples matching this amended query are returned by the DBMS (③). The integrity checking stage of verification is subsequently completed by comparing these tuples to the matching fingerprints that are kept on the blockchain (④, ⑤). Verification provides the end user with the results of the initial SQL query when the check has been successfully completed. As a result, the end user cannot see the integrity checking process and is only informed if the integrity check is unsuccessful. If INSERT, UPDATE, or DELETE requests contain nested SELECT queries, steps ② through ⑤ might need to be performed twice.

Algorithm 4 Query Student Marks Update History

Require: StudentID, Subject

Ensure: History of Marks Updates for the specified StudentID and Subject

```

1: Initialize Blockchain Network Components
2: Prepare Client Application with User Credentials
3: Create a Query Request
4: Set Query Inputs: (StudentID, Subject)
5: Send Query Request to the Ledger
6: Receive Query Results from the Ledger
7: if Query Results are not empty then
8:   Display the History of Marks Updates
9:   for each Transaction in Query Results do
10:    Extract Transaction Details:
11:    Print TransactionID, OldMarks, NewMarks, Timestamp, UpdateBy
12:   end for
13: else
14:   Notify User that No History Exists for the Given StudentID and Subject
15: end if

```

The algorithm 4 is used for querying the history of student mark updates on a blockchain. This algorithm outlines the process for retrieving and displaying the transaction

history related to a specific student's marks, including all past updates

IV. EXPERIMENTAL RESULTS

We have put our strategy into practice and tested it in the given circumstance. Imagine an educational setting where grades must be assigned to students. Every semester, the institution hosts a number of courses. There are several enrolled students and one or more staffs teaching each course.

Computation setup: : Hyperledger Fabric was deployed using docker version 27.0.3. We use Oracle as the underlying DBMS and the NodeJS programming language to construct our SQL parser. For testing, Hyperledger Fabric is deployed on one system, and queries and requests are parsed and sent to Hyperledger on another. The machines were networked together using a LAN cable.

Database: : Any relational database can be used to record the grades and enrollment information. Oracle has been utilized in our research. The database's tables are related to the following entities: staffs, students, and courses. The enrollment table, which contains data about students enrolled in several courses taught by a certain staff member, serves as a marker of their relationship. In our case, the enrollment table contains student grade information that needs to be kept private. Since only the staff is permitted to alter grades in our situation, as previously said, we have updated the enrollment table to add columns for the staff's identify and transaction ID. Oracle views are used to restrict which sections of the table are accessible to which entities. This is carried out in order to guard against any illegal access to the data kept in the database.

Web application: : To facilitate communication with the users, we create a web application. NodeJS has been utilized in the application's implementation. Every employee and student has a local web server that they use to retrieve and fill in data from the Oracle database by connecting with their username and password. While staff can alter a student's grade for any student, each student can only view his own grades.

Blockchain: : We are utilizing Hyperledger Fabric for a distributed blockchain because it offers a natural abstraction for use cases involving permissioned blockchains and concentrates on generic data retrieval, timestamping, and archiving rather than the exchange of assets between members. Installing Hyperledger and using our script to connect to a central Hyperledger node is what we expect from each participant. Next, an address for the chain is given to each member, which may be used to grant rights. Students can validate their grades in accordance with protocol because the stream's permissions are distributed based

on staff members' addresses that were obtained from public key infrastructure. This permits them to append transactions and mine blocks while permitting everyone to view the blockchain. The verification is conducted as follows.

Every time a student accesses his grades, the server retrieves data from the central database, which also includes the transaction id and identifier. The transaction id found in the row is used by the application to get the transaction from the log-stream. The public key is retrieved by the application using the identification from the public key infrastructure. After then, this public key is used to validate the transaction's signature. The student can view his marks if the signature is successfully validated and the data's hash matches the hash included in the transaction. If any of the aforementioned procedures are not followed, a notification is sent to the relevant authorities to investigate any unauthorized changes made to the database and an alarm is raised on the student's end.

The outcomes of every course the student completes are represented in the Table I. The information will first be stored on the blockchain and then externally as Oracle. The database has a feature called PERMISSIONGRANTED that lists the staffs and administrators who are authorized to modify the learning outcomes of their students. The most recent data operation on the course results is represented by the ACTION field, which is created when the smart contract is called. Its values include create (new course results), update (update course results), and change access (change the right to alter learning results).

The hash of the tuple (SUBJECT_MS, STUDENT_MS, TEACHER_MS, SCORE) is signed on the backend whenever a staff member inserts grades for any student (INSERT query), and the signature is broadcast on log-stream. In its internal log, the application keeps track of the actual inquiry. The application collects the grade's real data and pushes the query to the database after attaching the staff member and transaction identifiers once this transaction has been mined in a block and a predetermined number of confirmations have been obtained.

We perform 43 queries with various complexities on the information obtained from the training management system at Can Tho University. In comparison to Hyperledger lookup, we have found that the time required to conduct queries on Oracle is extremely small. There is a linear relationship between the overall query runtime and the number of tuples a query affects. The average Hyperledger lookup time for a SELECT query is 0.07–0.3 seconds per tuple, but queries that need to modify tuples, such INSERT, UPDATE, and DELETE, often take 0.9–3.1 seconds per tuple. The peer-consensus protocol, which requires adding a fingerprint or changing the number of tuples per table on

Table I
THE STRUCTURE OF BLOCKCHAIN DATA DESIGN.

Attribute name	Data type	Description
Result_Key	varchar(50)	Composite Key learning outcomes
ID	varchar(20)	ID of learning result
GROUP_MA	varchar(10)	Course group identification number
SUBJECT_MS	varchar(10)	Course identification number
STUDENT_MS	varchar(10)	Student identification number
STUDENT_NAME	varchar(20)	Student name
TEACHER_MS	varchar(10)	Staff identification number
SCORE	int	Course grades
PERMISSIONGRANTED	varchar(20)	List of score editors
ACTION	varchar(20)	Type of operation for the course results

the Hyperledger network, is the reason for the longer query times for the tuple updates. In addition, we also conducted 30 experiments by intentionally changing students' grades by directly interfering with the database. The system was able to detect all these unauthorized changes by comparing with the data stored on Hyperledger.

This thorough investigation shows that the average Hyperledger lookup time for a SELECT query is 0.08–0.1 seconds per tuple, but the average search time for queries that need to modify tuples, including INSERT, UPDATE, and DELETE, is roughly 0.8–2.5 seconds per tuple. The peer-consensus protocol's requirement to add a fingerprint or change the number of tuples per table on the Hyperledger network is the reason for the longer query times for tuple updates.

The findings suggest that using blockchains in conjunction with DBMSs leads in overheads that don't seem ideal for high-performance throughput. However, our goal is to be without transferring all of the DBMS data to a blockchain, and thus maintain privacy, whereas we identify an internal modification by using a blockchain's tamper-resistance features.

V. CONCLUSION

In this work, we have introduced a blockchain-based method for identifying internal modification and other unapproved database entry modifications. Because a hash value and a public key are stored in each row in addition to the regular data, the system that is being shown has a low storage overhead. Verification process can assign the modifications to the appropriate owner for any queries that alter the data that are processed through it. Because verification verifies each query's tuple integrity against the blockchain, there is some latency. However, we believe that this latency can be accepted rather than jeopardizing the integrity of the data in systems where the DBMS tuples' integrity is crucial since essential decisions are relied on them. For future work, we are going to improve the time of

data verification process in case of extremely large amount of data.

REFERENCES

- [1] M. Ubaka, A. Azeta, A. Oni, H. Okagbue, and N. Omoregbe, "Securing educational data using agent-based blockchain technology," *International Journal of Scientific & Technology Research*, vol. 9, pp. 2936–2938, 01 2020.
- [2] T. K. Agrawal, V. Kumar, R. Pal, L. Wang, and Y. Chen, "Blockchain-based framework for supply chain traceability: A case example of textile and clothing industry," *Computers and Industrial Engineering*, vol. 154, p. 107130, 2021.
- [3] R. Ghani, A. Al-karkhi, A. Khudhair, and L. Aljobouri, "Blockchain-based student certificate management and system sharing using hyperledger fabric platform," *Periodicals of Engineering and Natural Sciences (PEN)*, vol. 10, p. 207, 04 2022.
- [4] R. Awadallah and A. Samsudin, "Using blockchain in cloud computing to enhance relational database security," *IEEE Access*, vol. 9, pp. 137 353–137 366, 2021.
- [5] H. Wu, S. Jiang, and J. Cao, "High-efficiency blockchain-based supply chain traceability," 2022.
- [6] F. Casino, V. Kanakaris, T. K. Dasaklis, S. J. Moschuris, S. Stachtariis, M. Pagoni, and N. P. Rachaniotis, "Blockchain-based food supply chain traceability: a case study in the dairy sector," *International Journal of Production Research*, vol. 59, pp. 5758 – 5770, 2020.
- [7] S. Cao, H. Johnson, and A. Tulloch, "Exploring blockchain-based traceability for food supply chain sustainability: Towards a better way of sustainability communication with consumers," *Procedia Computer Science*, vol. 217, pp. 1437–1445, 2023, 4th International Conference on Industry 4.0 and Smart Manufacturing.
- [8] G. Varavallo, G. Caragnano, F. Bertone, L. Vernetti-Prot, and O. Terzo, "Traceability platform based on green blockchain: An application case study in dairy supply chain," *Sustainability*, vol. 14, no. 6, 2022.
- [9] S. I. M. Ali, H. Farouk, and H. Sharaf, "A blockchain-based models for student information systems," *Egyptian Informatics Journal*, vol. 23, no. 2, pp. 187–196, 2022.
- [10] X. Liang and S. Xu, "Student performance protection based on blockchain technology," *Journal of Physics: Conference Series*, vol. 1748, p. 022006, 01 2021.
- [11] I. M. Aldayflah, W. Zhao, H. Upadhyay, and L. Lagos, "The design and implementation of a secure datastore based on ethereum smart contract," *Applied Sciences*, vol. 13, no. 9, 2023.



Cu Vinh Loc received a degree in computer engineering from Can Tho University, Vietnam, in 2001. In 2009, He received his master's degree in Computer Science from the Asian Institute of Technology, Thailand. He received his PhD in Computer Science from the University of La Rochelle, France in 2019. He is currently a lecturer at Can Tho University. His research interests include data hiding, document analysis and recognition, natural language processing, blockchain.

Email: cvloc@ctu.edu.vn



Tran Hoang Viet got his first degree, Engineer in Information Technology, from CanTho University of Technology (CTU) in 2000 and had been working for the Faculty of Information Technology until 2001. He had become a researcher at Cantho University Software Center until 2010. Then, he was awarded a Master's scholarship from the MOET of Vietnam and studied at the HAN University of Applied Sciences, Netherlands until October 2011. Then, he was awarded a PhD scholarship from Soongsil University, Korea. His PhD research topic is related to "3D Stereoscopy Techniques in Subtitle, Face Detection, and Simulation Machine". He received the PhD degree in Media Engineering in August 2017. Since August 2017, he has been working as a researcher at the Cantho University Software Center. He became a lecturer at CIT in 2019 and has been a deputy director there since November 2021.

His research interests focus on Machine Learning, Deep Learning, Computer Vision, Computer Graphics, and Software Engineering. Email: thviet@ctu.edu.vn



Tran Thi Dinh Xuyen graduated from Can Tho University in 2010 with a degree in Information Technology. Currently, she is in charge of managing and operating the Monitoring and Operation Department (operating the Monitoring and Operation Center of Soc Trang province), the Digital Technology Center under the Department of Information and Communications of Soc Trang province.



Le Hoang Thao received a degree in computer engineering from Can Tho University, Vietnam, in 2000. In 2024, he received his master's degree in Computer Science from the Asian Institute of Technology, Thailand. He is currently a lecturer at Can Tho University and also the director of Can Tho University Software Center. His research interests includes IoT, Solutions for university digital transformation and E-government.

Email: lhthao@ctu.edu.vn



Truong Xuan Viet, a former Computer Science Engineer at Can Tho University, holds a Master's degree from Marne-La-Vallée University (France) and a Ph.D. from Paris 6 University (France). His research focuses on a diverse range of areas, including Geographic Information Systems (GIS), Remote Sensing, Agent-Based Simulation, Internet of Things (IoT), Networking, and Machine Learning. He has leveraged his knowledge to develop innovative solutions for Complex Systems, such as Smart Campus, Environmental Management, and Spatial Data Infrastructure systems.

Email: txviet@ctu.edu.vn



Truong Minh Thai is a former Bachelor of Engineering in Computer Science student at Can Tho University. He obtained a Master's in Information System Development from HAN University (2010), Netherlands, and his Doctorate in Computer Science from Toulouse 1 Capitole University (2015), France. In his research, he concentrates on Data Warehousing, Multi-Agent-Based Systems, Blockchain, and the Internet of Things (IoT), applying these technologies to enhance software development in the fields of agriculture and aquaculture.

Email: tmthai@cit.ctu.edu.vn

Integrating Self-Supervised Learning with Nonlinear Classifiers in Lightweight Swin Transformer for X-Ray Image Classification

Tri-Thuc Vo¹, Thanh-Nghi Do^{1,2}

¹ College of Information Technology, Can Tho University, 92000-Cantho, Vietnam

² UMI UMMISCO 209 (IRD/UPMC), Sorbonne University, Pierre and Marie Curie University - Paris 6, France

Correspondence: Thanh-Nghi Do, email: dtngi@ctu.edu.vn

Received: 30/07/2024, revised: 23/09/2024, accepted: 27/10/2024

Digital Object Identifier: 10.32913/mic-ict-research-vn.v2024.n3.1313

Abstract: In this paper, we present a new approach about the integration Self-Supervised Learning with nonlinear Classifiers in Lightweight Swin Transformer (*SSLnC-LSwiT*) for improving performance of X-ray image classification. Our approach leverages unlabeled data to address the issue of labeled data scarcity in the medical field by using self-supervised learning (SSL) to extract features. One of our key contributions is the introduction of the Lightweight SwiT architecture, a more lightweight variant of SwiT, designed to enhance computational efficiency, reduce model complexity, and shorten training time. To further improve classification efficiency, we propose the integration of a nonlinear classifier instead of a linear classifier in Lightweight SwiT. The experimental results underscore our contributions, demonstrating significant reductions in model training time and notable improvements in classification performance. Our proposed method, which integrates SSL based on LSwiT with a nonlinear LightGBM classifier, achieves an accuracy of up to 87%, improving by 1.8% over the non-LightGBM SwiT version and reducing training time significantly (3:23:00 vs. 7:37:29) compared to the original SwiT architecture.

Keywords: *Self-supervised learning, X-ray image, swin transformer, multi-class classification*

Tiêu đề: Tích hợp học tự giám sát với thuật toán phân lớp phi tuyến trong kiến trúc Lightweight Swin Transformer để phân lớp hình ảnh X-quang

Tóm tắt: Trong bài báo này, chúng tôi trình bày một phương pháp mới về tích hợp học tự giám sát (SSL) với thuật toán phân lớp phi tuyến trong kiến trúc Lightweight Swin Transformer (*SSLnC-LSwiT*) nhằm cải thiện hiệu quả phân lớp ảnh X-quang. Phương pháp của chúng tôi nhằm khai thác dữ liệu chưa được gán nhãn để giải quyết vấn đề khan hiếm dữ liệu có nhãn trong lĩnh vực y tế bằng cách tiếp cận học tự giám sát để học và trích xuất đặc trưng. Một trong những đóng góp quan trọng của chúng tôi là giới thiệu kiến trúc Lightweight SwiT, một biến thể của SwiT với kiến trúc đơn giản hơn, được đề xuất nhằm nâng cao hiệu quả tính toán, giảm độ phức tạp của mô hình và rút ngắn thời gian huấn luyện. Để cải thiện hiệu quả phân lớp, chúng tôi đề xuất tích hợp thuật toán phân lớp phi tuyến thay vì bộ phân lớp tuyến tính trong Lightweight SwiT. Kết quả thực nghiệm nhấn mạnh các đóng góp của chúng tôi, với sự giảm đáng kể thời gian huấn luyện mô hình và cải thiện đáng kể hiệu quả phân lớp. Phương pháp đề xuất, kết hợp SSL dựa trên LSwiT với thuật toán LightGBM, đạt độ chính xác 87%, tăng 1,8% so với phiên bản SwiT không sử dụng LightGBM và giảm đáng kể thời gian huấn luyện (3:23:00 so với 7:37:29) so với kiến trúc SwiT ban đầu.

Từ khóa: *Self-supervised learning, X-ray image, swin transformer, multi-class classification*

I. INTRODUCTION

The Covid-19 pandemic has caused global devastation, with millions of cases and deaths reported by the World Health Organization. Countries worldwide have faced overwhelmed healthcare systems and limited resources. In this context, chest X-ray (CXR) images have played a crucial

role in diagnosing and monitoring Covid-19 patients. CXR images are also essential in diagnosing lung diseases due to their ability to provide detailed images of lung structures. They primarily help detect abnormalities, enabling doctors to identify and assess lung conditions such as tumors, nodules, infections, or injuries. The X-ray method is preferred over other imaging methods for diagnosing lung diseases

because it is significantly more cost-effective than technologies like MRI or PET, making it a more economical choice for both patients and healthcare facilities. Additionally, radiographs are widely available and accessible at various healthcare settings, from large hospitals to small clinics. Despite their usefulness, diagnosing lung diseases with X-rays involves a risk of misdiagnosis, largely dependent on the skill and experience of the radiologist. Therefore, incorporating supportive technologies like artificial intelligence is crucial to enhance accuracy and reduce errors in X-ray image diagnosis.

The emergence of Convolutional Neural Networks (CNNs) has significantly contributed to computer vision, particularly in medical image classification and aiding disease diagnosis from medical images [1]. CNNs transformed image classification and recognition tasks by learning features through convolutional and fully connected layers, significantly improving diagnostic accuracy for conditions seen in X-rays and MRIs [2]. Vision Transformers (ViT) [3] introduced the self-attention mechanism, originally developed for natural language processing, to image processing. Swin Transformer (SwinT) [4] was designed to address the limitations of ViT, such as handling large images and improving computational efficiency. It designs a hierarchical structure to capture features at various levels, enhancing the analysis of complex image details. Both ViTs and SwinT have been extensively studied in medical image analysis [5]. However, supervised learning approaches using CNNs or ViT face significant drawbacks, primarily the need for large labeled datasets, which can be expensive and time-consuming to obtain.

Self-supervised learning is a machine learning method that leverages unlabeled data to learn features and patterns, enhancing the ability to understand and classify data without human intervention in labeling. This method has achieved significant milestones and increased efficiency in natural image classification [6]. It implements various backbones in its architecture, such as CNNs [7], ViT [8], and SwinT [9]. By effectively utilizing unlabeled data and fine-tuning it with a small labeled dataset, this approach provides notable assistance in the medical field [10], where labeled data is scarce and data labeling costs are high. In [11], Mathilde Caron et al. explored how SSL with ViT. DINO, a self-distillation method using a student-teacher framework, enables ViT-Base to achieve 80.1% top-1 accuracy on ImageNet. The authors proposed MoCo-CXR [12], an adaptation of Momentum Contrast (MoCo) for improving CXR pathology detection. Models using MoCo-CXR-pretrained representations outperform those without, enhancing AUC on the smaller Shenzhen tuberculosis dataset. The authors proposed CheSS, a SSL method for

CXR models, trained on a dataset of 4.8 million X-ray images [13]. CheSS achieved a 28.5% increase in accuracy for internal 6-class disease classification and a 1.3% mean AUC improvement on the CheXpert dataset, with an 11.4% increase using 1% of the data.

Our research introduces a new method to enhance CXR image classification while reducing model complexity and training time. The proposed Lightweight SwinT architecture (LSwinT), which is based on SwinT, significantly reduces training duration due to its simpler design, making it suitable for deployment in resource- and energy-constrained environments. For the goal of improving model performance, SSL is used with unlabeled CXR images to extract features from abundant unlabeled data, addressing the scarcity of labeled data in medical fields. Additionally, nonlinear classifiers are integrated instead of linear ones during the supervised learning phase with labeled data. Nonlinear classifiers effectively handle complex data and deliver high accuracy. The obtained results show that the nonlinear fine-tuned SSL pre-trained model outperforms the linear fine-tuned ImageNet pre-trained model. The findings also highlight that our proposed method, which leverages contributions from unlabeled X-ray images with SSL on LSwiT and the integration of a nonlinear classifier, enhances classification performance. Training time is significantly reduced with the LSwiT architecture combined classifiers compared to SwinT model.

The paper is organized as follows: Section II reviews prior research on lung disease classification using X-ray images. Section III describes our proposed method. Section IV details the experimental results, and Section V summarizes the findings and suggests future research directions.

II. RELATED WORK

In analysis of CXR images, recent research has focused on developing machine learning and deep learning methods to improve diagnostic accuracy for various pathologies. In [14], the Stanford University team developed CheXNeXt, a 121-layer DenseNet CNN for detecting 14 pathologies, including pneumonia and pleural effusion, in CXR images. Their algorithm matches board-certified radiologists in detecting multiple thoracic pathologies. Chen K.C et al. presented a diagnostic system for pediatric lung diseases using CXR images [15]. A YOLOv3 model crops the lung field, and three multi-classification methods were compared. The system detects abnormalities with 92.47% accuracy and identifies specific conditions with accuracies ranging from 71.94% to 85.71%. The study in [16] proposed an approach to improve multi-label CXR classification using the Chest X-ray 14 dataset. By combining visual vectors from a fine-tuned ConvNeXt network with semantic vectors from

BioBERT and using a dual-weighted metric loss function, the model achieved an average AUC score of 0.826.

For diagnosing Covid-19 from CXR images, the authors in [17] proposed the CNN-ELM framework, which combines lightweight parallel CNN feature extraction with the classification power of the extreme learning machine algorithm. It achieved 90.92% accuracy and a 96.93% AUC for 17 classes. The research in [18] aimed to detect diseases from CXR images of Covid-19 patients, pneumonia patients, and healthy individuals using a hybrid feature extraction network named D3SENET. Feature vectors extracted by CNN models were reduced using feature selection algorithms and classified using SVM. The study [19] trained classifiers using features from a fine-tuned Momentum Contrast model with Resnet backbones. The results showed an accuracy increase of 1.5% to 4.8% compared to those achieved by MoCo with only fine-tuned the linear layer. In [20], Zahid Ullah et al. proposed a multi-task semi-supervised learning (MTSSL) framework using publicly available data from pneumonia, lung opacity, and pleural effusion tasks with the CheXpert dataset. They integrated an adversarial autoencoder within their MTSSL framework to enhance feature learning and maximize the benefits of multi-task learning. The authors in [21] proposed the classification method of X-ray images with three steps: pre-processing, feature extraction using HOG and LBP, and classification with different classifiers. This method achieved 98% classification accuracy.

The Vision Transformer has been specifically highlighted for its application in X-ray image classification [22]. In [23], the authors proposed the Input Enhanced Vision Transformer (IEViT) that improved performance on CXR images with various pathologies. Experiments on four datasets (tuberculosis, pneumonia, Covid-19) showed that IEViT outperformed ViT, achieving F1-scores between 96.39% and 100%, with up to a +5.82% improvement over ViT. The study in [24] compared various optimization methods for ViT models in predicting lung diseases from a dataset of 19,003 CXR images. The optimization method with RAdam achieved 95.87% accuracy with ViT on balanced classes, while FastViT with NAdam reached 97.63% accuracy on imbalanced classes. In [25], the authors assessed interpretation methods for ViTs in CXR classification. Layerwise Relevance Propagation outperforms other methods, offering better insights into ViT learning. Lan Huang et al. proposed a method to improve ViT for CXR image analysis by adding a sliding window approach for better lesion detection, an attention region module, and a parallel network for patient metadata integration [26]. The model achieved an average AUC of 0.831, with sensitivity of 0.863, specificity of 0.821, and accuracy of 0.834 in diagnosing 14 diseases. In

[27], the authors introduced a hybrid workflow combining an ensemble five CNNs with a Transformer encoder for disease identification.

Self-supervised learning is seen as a promising approach with significant contributions to medical image classification, including X-rays, by leveraging large amounts of unlabeled data to learn features [10, 28]. In [29], CheXzero, a deep learning model via SSL developed by Stanford University's research team, automates the detection and classification of thoracic pathologies in CXR images. It outperforms fully supervised models in external validation by detecting three out of eight pathologies and generalizing well across various tasks and datasets. In [30], the authors introduced DINO-CXR, an adaptation of DINO using a vision transformer for CXR classification. It outperforms its compared methods in accuracy, AUC and F1 scores, and requires less labeled data. In [31], Jingfeng Yao et al. presented EVA-X, a SSL model that effectively captures semantic and geometric features from unlabeled X-rays. It excels in detecting over 20 chest diseases and performs well in 11 medical tasks, reducing the need for annotated data. In [32], the authors proposed a new self-supervised transfer learning approach for detecting Covid-19 from CXR images. The method outperformed six SSL methods and six pretrained DCNNs, achieving high scores: AUC 0.999 and four-class accuracy 0.953 on the Covid-19 CXR dataset.

III. METHOD OF X-RAY IMAGE CLASSIFICATION

1. MoBY Self-Supervised Learning

The MoBY method was introduced by Zhenda Xie et al. in [9] with the Swin Transformer (SwinT) as the backbone of the architecture. The SwinT architecture is hierarchically structured with varying transformer block numbers at each stage. Stage 1 uses 2 blocks to process 4×4 image patches, generating basic features. Stage 2, with 2 blocks, captures localized interactions in downsampled features. Stage 3 significantly expands in complexity with 6 blocks for high-level feature extraction from further downsampled maps. Stage 4, with 2 blocks, refines these features for final classification. MoBY combines the strengths of MoCo v2 [7] and BYOL [33]. The architecture of MoBY includes an online encoder and a target encoder. The target encoder consists of a backbone and a projector, while the online encoder is similarly designed but includes an additional predictor. The online encoder undergoes updates through gradient-based optimization, whereas the target encoder is updated by taking a moving average of the parameters from the online encoder. The architecture inherits several key elements from MoCo v2, including its momentum

design, key queue, and the use of the contrastive loss function InfoNCE, all of which contribute to robust feature representation learning. The loss function measures the similarity of image pairs, aiming to maximize the similarity between positive pairs and minimize it between negative pairs. Positive image pairs are formed by applying two data augmentation techniques on the same image, and the loss function aims to bring their representations closer together to minimize their distance. Negative image pairs comprise dissimilar images, and the loss function maximizes their distance. Additionally, MoBY incorporates aspects from BYOL, such as the use of asymmetric encoders, a variety of data augmentation techniques, and a momentum scheduler.

In this paper, we propose a new approach with SSL based on a simplified architecture of SwinT. Our goal is to leverage the strengths of SSL to extract features from unlabeled medical data, particularly X-ray images. Given the limited dataset and training resources, we propose new architectures based on the SwinT architecture by reducing the number of blocks in each stage. We introduce Lightweight SwinT (LSwinT) with two versions: LSwiT1, with 1 block per stage, and LSwiT2, with 2 blocks per stage. Reducing the model size enhances computational efficiency. This approach reduces both training and inference times, making it suitable for resource-constrained environments.

2. Self-Supervised Learning with nonlinear Classifiers in Lightweight Swin Transformer (SSLnC-LSwinT)

The algorithm *SSLnC-LSwinT* (as illustrated in Fig. 1 and Algorithm 1) unfolds in a structured sequence comprising three distinct stages. Initially, in the SSL phase, the model is set up with Enc_{online} and Enc_{target} , both based on backbone like the LSwiT architecture. During training, the model processes the unlabeled dataset $\mathcal{D}_{unlabeled}$ where data augmentation techniques such as cropping, flipping, jitter, grayscale, and GaussianBlur, are employed to generate positive pairs. The *InfoNCE* loss function guides the updates for Enc_{online} , while Enc_{target} is refined using a momentum-based approach with a parameter m . The parameters for this step are set to 300 epochs, learning rate (0.001), weight decay (0.05), and momentum (0.99). This results in the creation of $Model_1$. In the subsequent fine-tuning stage, $Model_1$ is utilized with its backbone frozen to train a linear classifier on the labeled datasets, which include $\mathcal{D}_{train}$ and $\mathcal{D}_{valid}$. The performance of this fine-tuned model, referred to as $Model_2$, is then evaluated its effectiveness through $\mathcal{D}_{test}$. Instead of freezing Backbone $Model_1$ and training the classifier layer, we extract its features and train a nonlinear classifier. The nonlinear classifier is integrated as a replacement for the linear classifier in the Lightweight Swin Transformer, allowing us to leverage its capabilities

Algorithm 1: Self-supervised learning with nonlinear classifiers in LSwiT

```

1 Inputs:  $\mathcal{D}_{unlabeled}$ ,  $\mathcal{D}_{labeled}$ : ( $\mathcal{D}_{train}$ ,  $\mathcal{D}_{valid}$ ,  $\mathcal{D}_{test}$ ),
   Classifiers: (LightGBM, XGBoost, CatBoost),  $m$ :
   momentum
2 Outputs:  $Model_1$ ,  $Model_2$ , classifier models
3 begin
4   Phase 1: Self-Supervised Learning
5   Initialize model
6    $Enc_{online}$ 
7    $Enc_{target}$ 
8   for each epoch do
9     for each minibatch in  $\mathcal{D}_{unlabeled}$  do
10       Update  $Enc_{online}$ 
11       Update  $Enc_{target}$  using momentum:
12        $Enc_{target} \leftarrow m \cdot Enc_{target} + (1 - m) \cdot Enc_{online}$ 
13     end
14   end
15   Output 1:  $Model_1$ 
16   Phase 2: Supervised Learning
17   Load  $Model_1$  from Phase 1
18   Fine-tuning
19     Freeze  $Model_1$  backbone
20     Train the linear classifier on  $\mathcal{D}_{labeled}$ 
21   Output 2:  $Model_2$ 
22   Phase 3: Integrating Nonlinear Classifiers
23   Feature extraction
24    $Model'_2 \leftarrow Model_2$  removed the last layer
25   for each set  $\in \{\mathcal{D}_{train}, \mathcal{D}_{valid}, \mathcal{D}_{test}\}$  do
26      $\mathcal{D}'(features, labels) \leftarrow Model'_2(set)$ 
27   end
28   Training nonlinear classifiers
29   for each classifier  $clf \in Classifiers$  do
30     Train  $clf$  on  $\mathcal{D}'(features, labels)$ 
31   end
32   Output 3: classifier models
33 end

```

to enhance classification efficiency. The final stage focuses on integrating a nonlinear classifier. In this step, $Model_2$ is used to extract features from the labeled datasets $\mathcal{D}_{train}$, $\mathcal{D}_{valid}$, and $\mathcal{D}_{test}$ by removing its final linear layer. Various nonlinear classifiers, including *LightGBM*, *XGBoost*, and *CatBoost*, are then trained on these extracted features with learning rate (0.23) and depth (8). The performance of each classifier is evaluated using $\mathcal{D}_{test}$.

3. Nonlinear classifiers

LightGBM, short for Light Gradient Boosting Machine, is a highly efficient gradient boosting framework developed

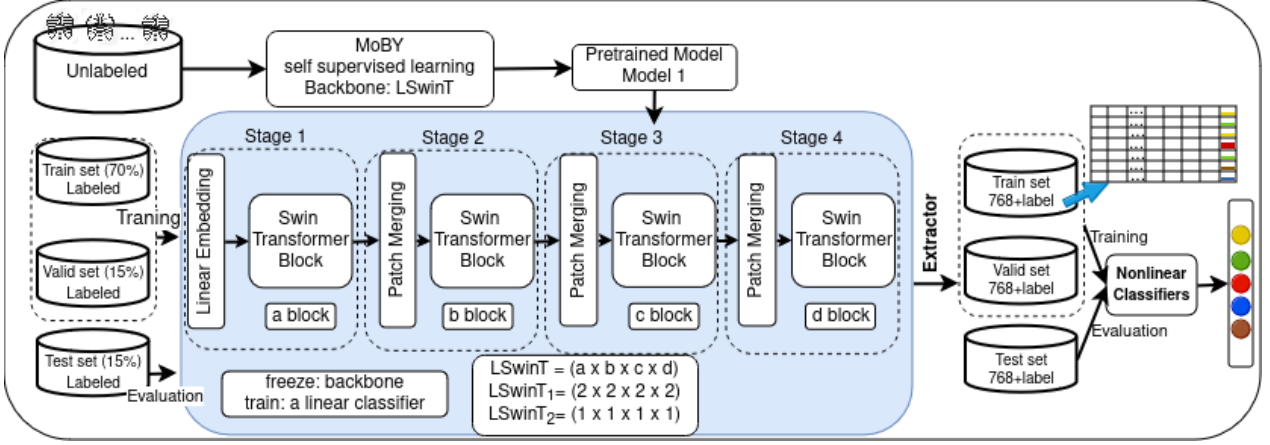


Figure 1. Diagram of *SSLnC-LSwinT* for chest X-ray image classification.

by Microsoft [34]. It is designed to be fast, distributed, and capable of handling large datasets with high accuracy. LightGBM used a novel technique called Gradient-based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB) to significantly improve the training speed and reduce memory usage. GOSS selectively retains instances with large gradients to maintain the accuracy, while EFB bundles mutually exclusive features to reduce the number of features. This results in faster computation without compromising model performance.

XGBoost [35], which stands for Extreme Gradient Boosting, is a highly efficient machine learning algorithm optimized for speed and performance. Developed by Tianqi Chen, XGBoost enhances traditional gradient boosting by incorporating a variety of advanced features. These include regularization to prevent overfitting, parallel processing to speed up computations. XGBoost also supports tree pruning and built-in cross-validation, which contribute to its robust performance. The algorithm can handle large datasets with high dimensionality and is highly scalable, making it suitable for complex tasks.

CatBoost [36] is an effective gradient boosting algorithm developed by Yandex, optimized for efficiently handling categorical features. Unlike traditional gradient boosting methods that often require preprocessing of categorical data, CatBoost incorporates categorical variables directly into its training process. This method helps to prevent overfitting by using permutation-driven target statistics for categorical features. CatBoost also leverages symmetric trees and gradient-based optimization to enhance model accuracy and reduce training time.

4. Chest X-Ray image data set

Our study about *SSLnC-LSwinT* used two datasets: an unlabeled one from the CheXpert dataset, containing

Table I
DATASET OF CHEST X-RAY IMAGES.

Label	Trainset	Validset	Testset
Normal	18,425	3,949	3,948
Covid-19	14,252	3,054	3,054
Edema	23,240	4,980	4,980
Mass-nodule	4,077	873	874
Pneumothorax	9,303	1,993	1,994

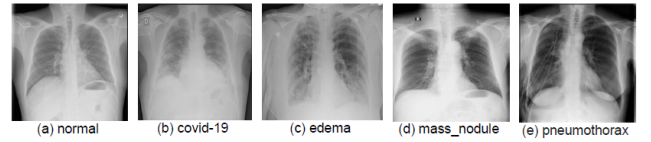


Figure 2. Sample of chest x-ray images for each class

over 200,000 X-ray images, with 120,000 images used for training the MoBY model, and a labeled dataset with 98,996 images across five classes (normal, Covid-19, edema, mass nodule, pneumothorax) from various sources as in [19]. Both datasets were resized to 224x224 pixels. The labeled dataset was divided into three subsets that include training (70%), validation (15%), and test (15%). Details are in Table I, and visual examples are shown in Fig. 2.

IV. EXPERIMENTAL RESULTS

A Python program was developed for *SSLnC-LSwinT* in CXR image classification, utilizing SSL and fine-tuning with SwinT and LSwiT architectures. Scikit-learn [37] and PyTorch [38] were used for implementation. Experiments were conducted on a computer running Ubuntu 22.04.3, equipped with an AMD® Ryzen 9 5900x 12-core processor, 32 GB of RAM, and an NVIDIA GeForce GTX 1080 Ti

Table II
ACCURACY AND TRAINING TIME FOR DIFFERENT MODELS.

Model	Accuracy (%)	Training Time (hh:mm:ss)
<i>MI-SwinT</i>	81.8	07:31:47
MI-SwinT+LightGBM	85.3	07:34:52
MI-SwinT+CatBoost	84.2	08:59:30
MI-SwinT+XGBoost	85.0	07:45:29
<i>MS-SwinT</i>	85.2	07:37:29
MS-SwinT+LightGBM	87.2	07:40:35
MS-SwinT+CatBoost	86.4	09:05:08
MS-SwinT+XGBoost	87.0	07:51:09
<i>MS-LSwinT1</i>	82.6	03:20:00
MS-LSwinT1+LightGBM	87.0	03:23:00
MS-LSwinT1+CatBoost	86.5	04:49:04
MS-LSwinT1+XGBoost	87.1	03:34:18
<i>MS-LSwinT2</i>	84.1	05:59:02
MS-LSwinT2+LightGBM	87.0	06:02:03
MS-LSwinT2+CatBoost	86.3	07:13:35
MS-LSwinT2+XGBoost	86.9	06:12:58

featuring 11 GB of GDDR5X memory and 3584 CUDA cores.

The SwinT architecture comprises four stages with the following number of blocks: Stage 1 has 2 blocks, Stage 2 has 2 blocks, Stage 3 has 6 blocks, and Stage 4 has 2 blocks. Our proposed LSwinT architecture has two versions: LSwinT1, with 1 block per stage, and LSwinT2, with 2 blocks per stage. MI-SwinT is fine-tuned from the ImageNet pretrained model using the SwinT architecture, while MS-SwinT is fine-tuned from a SSL model on unlabeled X-ray images using the same architecture. MS-LSwinT1 and MS-LSwinT2 are fine-tuned from a pretrained MoBY model with different architectures; MS-LSwinT1 uses the LSwinT1 architecture, and MS-LSwinT2 uses the LSwinT2 architecture.

The experimental results of the four models (MI-SwinT, MS-SwinT, MS-LSwinT1, and MS-LSwinT2) are shown in Tab. II and Fig. 3. MS-SwinT achieves the highest performance with an accuracy of 85.2%, outperforming MI-SwinT, which has an accuracy of 81.8%. This indicates that SSL on unlabeled X-ray images provides a significant advantage over pre-training with ImageNet for fine-tuning the linear classifier. MS-LSwinT1, with an accuracy of 82.6%, performs slightly better than MI-SwinT, highlighting the effectiveness of the LSwinT architecture even with fewer blocks. MS-LSwinT2, achieving an accuracy of 84.1%, also demonstrates superior performance compared to MI-SwinT but lower than MS-SwinT.

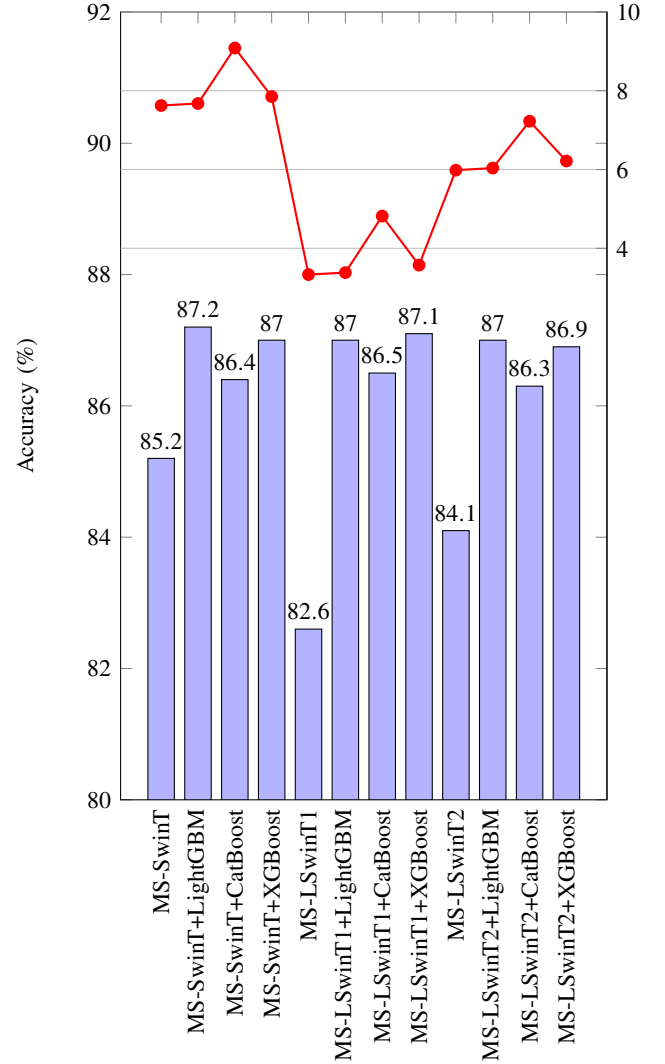


Figure 3. Accuracy and Training time for different models.

Integrating MS-LSwinT1 and MS-LSwinT2 as feature extractors with various nonlinear classifiers results in notable performance improvements in accuracy, showing no difference compared to MS-SwinT combined with nonlinear classifiers. Combining MS-SwinT with a classifier, LightGBM achieves the highest accuracy at 87.2%. XGBoost comes close with 87%, while CatBoost shows a slightly lower accuracy of 86.4%. When MS-LSwinT1 is paired with nonlinear classifiers, accuracy improves substantially. XGBoost performs well, reaching 87.1%, followed by LightGBM at 87.0% and CatBoost at 86.5%. Similarly, MS-LSwinT2's accuracy is markedly improved when combined with nonlinear classifiers, with the highest increase observed using LightGBM (+2.9%), followed by XGBoost (+2.8%), and CatBoost (+2.2%). These findings underscore that both MS-LSwinT1 and MS-LSwinT2, when paired with nonlinear classifiers, lead to enhanced feature

utilization and superior classification performance.

Training time for 100 epochs of fine-tuning a linear classifier reveals distinct differences among these models (see Tab. II and Fig. 3 with the red line). MS-SwinT and MS-LSwiT, both utilizing the SwiT architecture, have nearly identical training times of 07:31:47 and 07:37:29, respectively. In contrast, the LSwiT architectures demonstrate significantly shorter training times, with MS-LSwiT1 completing in 03:20:00 and MS-LSwiT2 in 05:59:02. This difference highlights the efficiency of LSwiT architectures, likely due to their lightweight design, which reduces computational complexity and speeds up training and inference. The LSwiT models are particularly beneficial in resource-limited scenarios.

The training time reveals notable variations across models combined with boosting algorithms. For MS-SwinT, there are slight increases in training time when combined with LightGBM (07:40:35) and XGBoost (07:51:09). However, CatBoost significantly extends the training time to 09:05:08. Comparing the lightweight models, MS-LSwiT1 exhibits the shortest training time of 03:20:00. Incorporating boosting algorithms results in minimal time increases for LightGBM (03:23:00) and XGBoost (03:34:18), but a substantial increase with CatBoost, reaching 04:49:04. Similarly, MS-LSwiT2 requires 05:59:02 to train, with slight time increases for LightGBM (06:02:03) and XGBoost (06:12:58), while CatBoost again results in the longest training time at 07:13:35.

The results show that combining MS-LSwiT1 and MS-LSwiT2 with classifiers, such as LightGBM, CatBoost, and XGBoost, leads to improved performance, aligning closely with MS-SwinT combined with the same classifiers. For example, MS-LSwiT1 with LightGBM achieves 87% accuracy, closely matching the 87.2% of MS-SwinT with LightGBM. MS-LSwiT1+XGBoost reaches 87.1%, which is nearly identical to MS-SwinT+XGBoost's 87%. Similarly, MS-LSwiT2 with these classifiers shows comparable performance, with MS-LSwiT2 using LightGBM and XGBoost achieving 87% and 86.9%, respectively. Combining MS-LSwiT1 and MS-LSwiT2 with classifiers shows no performance difference compared to MS-SwinT with the same classifiers, while significantly reducing training time. Specifically, MS-LSwiT1 using LightGBM trains in 03:23:00, over four hours faster than MS-SwinT with LightGBM, which takes 07:40:35. Similarly, MS-LSwiT1 with XGBoost finishes in 03:34:18, compared to 07:51:09 for MS-SwinT with XGBoost. Even MS-LSwiT2 shows time reductions, with training times of 06:02:03 using LightGBM and 06:12:58 with XGBoost, compared to 07:40:35 and 07:51:09 for the MS-SwinT model. These results highlight the efficiency of the lightweight architec-

tures combined with nonlinear classifiers, delivering similar accuracy with significantly reduced training times.

The combination of MS-LSwiT1 with different classifiers demonstrates the accuracy improvement compared to MS-SwinT. Specifically, MS-LSwiT1 with LightGBM shows an accuracy increase of 1.8%, reaching 87%, compared to the 85.2% accuracy of MS-SwinT. Similarly, the combination of MS-LSwiT1 with CatBoost results in a 1.3% improvement. The most significant increase is observed with MS-LSwiT1 and XGBoost, which delivers a 1.9% increase, resulting in an accuracy of 87.1%. These improvements highlight the effectiveness of integrating lightweight models with advanced classifiers to enhance overall performance.

V. CONCLUSION AND FUTURE WORKS

In this paper, we introduce an innovative approach that combines Self-Supervised Learning with nonlinear classifiers in the Lightweight SwiT Transformer (SSLnCLSwiT) to enhance X-ray image classification performance. By leveraging unlabeled data, our method addresses labeled data scarcity in the medical field through SSL to learn features. We present the LSwiT architecture, a lightweight variant of SwiT, designed to improve computational efficiency, reduce model complexity, and shorten training time. Integrating a nonlinear classifier instead of a linear one further boosts classification efficiency. The study results demonstrate the effectiveness of our proposed approach. Fine-tuning a linear classifier for 100 epochs, MS-SwinT model takes 07:37:29, while LSwiT architectures are faster: MS-LSwiT1 at 03:20:00 and MS-LSwiT2 at 05:59:02. Results show that using MS-LSwiT1 with nonlinear classifiers enhances accuracy as MS-SwinT but with significantly reduced training time. The base MS-LSwiT1 model achieves 82.6% accuracy, which increases to 87.1% with XGBoost. The XGBoost provides the largest boost at +1.9%, followed by LightGBM at +1.8% compared to MS-SwinT.

Future work may incorporate clinical metadata to improve model performance and explore SSL techniques for on clinical metadata.

REFERENCES

- [1] S. K. Zhou, H. Greenspan, and D. Shen, *Deep learning for medical image analysis*. Academic Press, 2023.
- [2] K. Sailunaz, T. Özyer, J. Rokne, and R. Alhajj, "A survey of machine learning-based methods for covid-19 medical image analysis," *Medical & Biological Engineering & Computing*, vol. 61, no. 6, pp. 1257–1297, 2023.
- [3] A. Dosovitskiy *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," 2021. [Online]. Available: <https://arxiv.org/abs/2010.11929>

- [4] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," 2021. [Online]. Available: <https://arxiv.org/abs/2103.14030>
- [5] K. He, C. Gan, Z. Li, I. Rekik, Z. Yin, W. Ji, Y. Gao, Q. Wang, J. Zhang, and D. Shen, "Transformers in medical image analysis," *Intelligent Medicine*, vol. 3, no. 1, pp. 59–78, 2023.
- [6] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 9729–9738.
- [7] X. Chen, H. Fan, R. Girshick, and K. He, "Improved baselines with momentum contrastive learning," *arXiv preprint arXiv:2003.04297*, 2020.
- [8] X. Chen, S. Xie, and K. He, "An empirical study of training self-supervised vision transformers," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 9640–9649.
- [9] Z. Xie, Y. Lin, Z. Yao, Z. Zhang, Q. Dai, Y. Cao, and H. Hu, "Self-supervised learning with swin transformers," 2021. [Online]. Available: <https://arxiv.org/abs/2105.04553>
- [10] S.-C. Huang, A. Pareek, M. Jensen, M. P. Lungren, S. Yeung, and A. S. Chaudhari, "Self-supervised learning for medical image classification: a systematic review and implementation guidelines," *NPJ Digital Medicine*, vol. 6, no. 1, p. 74, 2023.
- [11] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 9650–9660.
- [12] H. Sowrirajan, J. Yang, A. Y. Ng, and P. Rajpurkar, "Moco pretraining improves representation and transferability of chest x-ray models," in *Medical Imaging with Deep Learning*. PMLR, 2021, pp. 728–744.
- [13] K. Cho *et al.*, "Chess: Chest x-ray pre-trained model via self-supervised contrastive learning," *Journal of Digital Imaging*, vol. 36, no. 3, pp. 902–910, 2023.
- [14] P. Rajpurkar and *et al.*, "Deep learning for chest radiograph diagnosis: A retrospective comparison of the chexnext algorithm to practicing radiologists," *PLoS medicine*, vol. 15, no. 11, p. e1002686, 2018.
- [15] K.-C. Chen and *et al.*, "Diagnosis of common pulmonary diseases in children by x-ray images and deep learning," *Scientific reports*, vol. 10, no. 1, p. 17374, 2020.
- [16] Y. Jin, H. Lu, W. Zhu, and W. Huo, "Deep learning based classification of multi-label chest x-ray images via dual-weighted metric loss," *Computers in Biology and Medicine*, vol. 157, p. 106683, 2023.
- [17] M. Nahiduzzaman and *et al.*, "Parallel cnn-elm: A multiclass classification of chest x-ray images to identify seventeen lung diseases including covid-19," *Expert Systems with Applications*, vol. 229, p. 120528, 2023.
- [18] M. Kaya and M. Eris, "D3senet: A hybrid deep feature extraction network for covid-19 classification using chest x-ray images," *Biomedical signal processing and control*, vol. 82, p. 104559, 2023.
- [19] T.-T. Vo and T.-N. Do, "Improving chest x-ray image classification via integration of self-supervised learning and machine learning algorithms," *Journal of Information & Communication Convergence Engineering*, vol. 22, no. 2, 2024.
- [20] Z. Ullah, M. Usman, and J. Gwak, "Mtss-aae: Multi-task semi-supervised adversarial autoencoding for covid-19 detection based on chest x-ray images," *Expert Systems with Applications*, vol. 216, p. 119475, 2023.
- [21] S. U. Amin, S. Taj, A. Hussain, and S. Seo, "An automated chest x-ray analysis for covid-19, tuberculosis, and pneumonia employing ensemble learning approach," *Biomedical Signal Processing and Control*, vol. 87, p. 105408, 2024.
- [22] S. Taslimi, S. Taslimi, N. Fathi, M. Salehi, and M. H. Rohban, "Swinchex: Multi-label classification on chest x-ray images with transformers," *arXiv preprint arXiv:2206.04246*, 2022. [Online]. Available: <https://arxiv.org/abs/2206.04246>
- [23] G. I. Okolo, S. Katsigiannis, and N. Ramzan, "Ievit: An enhanced vision transformer architecture for chest x-ray image classification," *Computer Methods and Programs in Biomedicine*, vol. 226, p. 107141, 2022.
- [24] J. Ko, S. Park, and H. G. Woo, "Optimization of vision transformer-based detection of lung diseases from chest x-ray images," *BMC Medical Informatics and Decision Making*, vol. 24, no. 1, p. 191, 2024.
- [25] P. Komorowski, H. Baniecki, and P. Biecek, "Towards evaluating explanations of vision transformers for medical imaging," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 3726–3732.
- [26] L. Huang, J. Ma, H. Yang, and Y. Wang, "Research and implementation of multi-disease diagnosis on chest x-ray based on vision transformer," *Quantitative Imaging in Medicine and Surgery*, vol. 14, no. 3, p. 2539, 2024.
- [27] C. C. Ukwuoma *et al.*, "A hybrid explainable ensemble transformer encoder for pneumonia identification from chest x-ray images," *Journal of Advanced Research*, vol. 48, pp. 191–211, 2023.
- [28] B. VanBerlo, J. Hoey, and A. Wong, "A survey of the impact of self-supervised pretraining for diagnostic tasks in medical x-ray, ct, mri, and ultrasound," *BMC Medical Imaging*, vol. 24, no. 1, p. 79, 2024.
- [29] E. Tiu, E. Talus, P. Patel, C. P. Langlotz, A. Y. Ng, and P. Rajpurkar, "Expert-level detection of pathologies from unannotated chest x-ray images via self-supervised learning," *Nature Biomedical Engineering*, vol. 6, no. 12, pp. 1399–1406, 2022.
- [30] M. Shakouri, F. Iranmanesh, and M. Eftekhari, "Dino-cxr: A self supervised method based on vision transformer for chest x-ray classification," in *International Symposium on Visual Computing*. Springer, 2023, pp. 320–331.
- [31] J. Yao, X. Wang, Y. Song, H. Zhao, J. Ma, Y. Chen, W. Liu, and B. Wang, "Eva-x: A foundation model for general chest x-ray analysis with self-supervised learning," 2024. [Online]. Available: <https://arxiv.org/abs/2405.05237>
- [32] G. Li, R. Togo, T. Ogawa, and M. Haseyama, "Covid-19 detection based on self-supervised transfer learning using chest x-ray images," *International Journal of Computer Assisted Radiology and Surgery*, vol. 18, no. 4, pp. 715–722, 2023.
- [33] J.-B. Grill *et al.*, "Bootstrap your own latent: A new approach to self-supervised learning," 2020. [Online]. Available: <https://arxiv.org/abs/2006.07733>
- [34] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," *Advances in neural information processing systems*, vol. 30, 2017.
- [35] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–794.
- [36] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "Catboost: unbiased boosting with categorical features," *Advances in neural information processing systems*, vol. 31, 2018.
- [37] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in python," *the Journal of machine Learning research*, vol. 12, 2012.

pp. 2825–2830, 2011.

- [38] A. Paszke *et al.*, “Pytorch: An imperative style, high-performance deep learning library,” 2019. [Online]. Available: <https://arxiv.org/abs/1912.01703>



Tri-Thuc Vo received the B.Eng. degree in Software Engineering from the Cantho University, Vietnam, in 2011. He received his MSc. degree in informatics from the University of Brest, France, in 2018. He is currently a lecturer at the College of Information Technology, Cantho University, Vietnam. His research interests include

medical data analysis and machine learning.

Email: vtthuc@ctu.edu.vn



Thanh-Nghi Do received his PhD. degree in informatics from the University of Nantes, France, in 2004. He is currently an associate professor at the College of Information Technology, Cantho University, Vietnam. He is also an associate researcher at UMI UMMISCO 209 (IRD/UPMC), Sorbonne University, and the Pierre and Marie

Curie University, France. His research interests include data mining with support vector machines, kernel-based methods, decision tree algorithms, ensemble-based learning, and information visualization. He has served on the program committees of international conferences and is a reviewer for journals in his fields of expertise. Email: dtngchi@ctu.edu.vn

Rice Leaf Disease Classification Using Knowledge Distillation

Md. Saiful Niaz¹, Tasmia Hassan Shabonty¹, Ashfaque Rahman Saad¹, Lucky Talukder Tarin¹, Dewan Md. Farid¹, and Huu-Hoa Nguyen²

¹ Department of Computer Science and Engineering, United International University United City, Madani Avenue, Badda, Dhaka 1212, Bangladesh

² College of Information and Communication Technology, Can Tho University, 92000, Can Tho, Vietnam

Correspondence: Huu-Hoa Nguyen, email: nhhoa@ctu.edu.vn

Received: 31/07/2024, revised: 23/09/2024, accepted: 06/11/2024

Digital Object Identifier: 10.32913/mic-ict-research-vn.v2024.n3.1310

Abstract: Rice is essential for global food security, particularly in Asian countries where it is a staple food and a significant contributor to the economy. Rice cultivation faces numerous challenges, including a variety of illnesses that can significantly impact the yield and quality of the harvest. Conventional disease detection methods are slow, require a lot of manual effort, and are prone to mistakes, emphasizing the critical requirement for automated approaches. This research explores the implementation of knowledge distillation, a method of transferring expertise from advanced Models to more basic ones, for identifying diseases in rice. We condensed the expertise of complex Models by training them extensively, creating a streamlined and effective Model. We employed DenseNet121 as the Teacher Model, and the Student Model is equipped with dual pipeline Models to address overfitting. Both of these Models passed through Quantization-Aware Training (QAT). The Student-Teacher collaborative Model achieved a precision of 98.47%, using significantly fewer parameters (5.9 million) than the Teacher-only Model. Finally, the whole Model was quantized to an 8-bit integer from 32-bit float points. This leads to a fairer method for effectively and accurately isolating small and complex characteristics. This small-scale is ideal for use on portable gadgets in environments with restricted resources, enabling farmers to quickly identify diseases for better crop management and increased worldwide food security measures.

Keywords: Deep learning, knowledge distillation, quantization, rice leaf disease classification.

Tiêu đề: Phân loại bệnh lá lúa bằng phương pháp chưng cất kiến thức

Tóm tắt: Lúa đóng vai trò thiết yếu đối với an ninh lương thực toàn cầu, đặc biệt tại các quốc gia châu Á, nơi lúa là thực phẩm chính và nguồn kinh tế quan trọng. Việc trồng lúa đối mặt với nhiều thách thức, bao gồm các loại bệnh có thể gây ảnh hưởng nghiêm trọng đến năng suất và chất lượng mùa màng. Các phương pháp phát hiện bệnh truyền thống thường chậm, tốn nhiều công sức và dễ xảy ra sai sót, do đó cần thiết phải có các giải pháp tự động hóa. Nghiên cứu này tìm hiểu việc áp dụng phương pháp chưng cất kiến thức – phương pháp chuyển giao kiến thức từ các mô hình phức tạp sang các mô hình đơn giản hơn – nhằm nhận diện bệnh trong cây lúa. Chúng tôi đã tối giản hóa kiến thức từ các mô hình phức tạp bằng cách huấn luyện chuyên sâu, tạo ra một mô hình hiệu quả và gọn nhẹ. DenseNet121 được sử dụng làm mô hình giáo viên, và mô hình học sinh được trang bị hệ thống đường dẫn kép để giảm thiểu tình trạng quá khớp. Cả hai mô hình đều trải qua quá trình huấn luyện nhận thức lượng tử (QAT). Mô hình hợp tác giữa giáo viên và học sinh đạt độ chính xác 98,47%, với số lượng tham số ít hơn đáng kể (5,9 triệu) so với mô hình chỉ dùng giáo viên. Cuối cùng, toàn bộ mô hình được lượng tử hóa từ số thực 32-bit sang số nguyên 8-bit, tạo ra một phương pháp hiệu quả hơn để tách biệt các đặc tính nhỏ và phức tạp một cách chính xác. Mô hình nhỏ gọn này lý tưởng cho các thiết bị di động trong môi trường hạn chế tài nguyên, giúp nông dân nhận diện bệnh nhanh chóng, hỗ trợ quản lý cây trồng tốt hơn và tăng cường an ninh lương thực toàn cầu.

Từ khóa: Deep learning, knowledge distillation, quantization, rice leaf disease classification.

I. INTRODUCTION

Rice is an important type of food that feeds billions of people all over the world. It is a significant aspect of Asian

culture, history, and economy as well as a main food source [1]. In Bangladesh, rice is essential for everyone's daily life and for how the country makes money. Almost 90% of the population here depends on rice for their calories [2]. In

the agriculture of Bangladesh, rice production contributes 40% to the country's GDP [2]. However, growing rice is not always easy. When plants get sick, they can't grow well and don't give as much rice. There are different diseases like bacterial leaf blight, blasts, sheath blight, and brown spots that can make this happen. If farmers don't find good ways to deal with these diseases, they can lose a lot of crops [3]. These diseases can also make the rice grains not as good and safe to eat. Rice disease causes global grain loss of 10% to 30% annually [4]. In traditional agriculture, identifying rice diseases typically depends on the experience gained by farmers over time. This method requires a lot of skilled people as well as significant time, effort, and money. However, it often leads to biased judgments and significant errors. As a result, it becomes difficult to correctly detect diseases, which makes it simple to miss the best window of opportunity for disease prevention and control. Moreover, keeping up with the need for real-time monitoring and predicting various diseases across a wide spectrum is quite difficult [5]. Hence, investigating automated techniques to identify common rice illnesses is urgently needed [6]. Thus, although rice is Bangladesh's main food source, it is equally critical to monitor the spread of these diseases and discover measures to contain them. It is crucial to identify rice diseases quickly and accurately in order to ensure rice production and protect global food security [7].

Deep learning technologies have taken revolutionary steps in many fields, and as a result, their potential for detecting agricultural diseases has been gradually investigated. When it comes to rice plant disease detection, deep learning and machine learning have significantly improved. These technologies work by using really smart computers that can look at lots and lots of pictures or data from fields to spot signs of diseases before they become a big problem [8]. Farmers can therefore take prompt action to prevent the illnesses from spreading and damaging their crops. To solve this problem machine learning and convolutional neural networks have implemented different new techniques and obtained very good results as well [9]. Knowledge distillation involves transferring knowledge from a complex model (teacher) to a simpler model (student). In paddy disease detection, this means training a large, resource-intensive model to identify diseases accurately, and then distilling its knowledge into a smaller, more efficient model suitable for deployment in resource-constrained environments like rural areas. This technique minimises computing resources as well as maintains performance. By placing these lightweight models on low-power devices, like cell-phones or edge devices, farmers can acquire reliable disease detection tools even in the absence of internet connectivity, enabling timely interventions to safeguard their crops and

livelihood.

Distilling knowledge from a big, complex model to a smaller one works well, even when data is limited. Distillation can condense the benefits of ensemble learning into a single, deployable model, like for Android voice search. It's possible to improve the performance of large networks by training specialist nets [10]. By using the knowledge distillation technique, Musa et al. [11] proposed a new method for detecting plant diseases in smart hydroponic systems. This system uses a special type of deep learning model designed to be less powerful and use less energy. This is achieved by using a simpler network structure and a technique called knowledge distillation. The authors found that their model achieved a high accuracy of 99.4% in detecting diseases while significantly reducing power consumption compared to existing models. Hu et al. [12] used knowledge distillation techniques to detect maize leaf diseases using a lightweight image recognition model. This model is based on an improved version of YOLOv5s which is a popular detection system.

In the proposed methodology, we have leveraged knowledge distillation from larger models to function as mentors in instructing pared-down models with reduced parameters. This methodology necessitated the training of numerous high-capacity models such as Google's Inception, MobileNet, and Xception, as well as the VGG-19, DenseNet-121, and NasNetmobile architectures. We obtained the DenseNet-121 model, which demonstrated the highest level of accuracy, and opted to utilise it for training our student model. We amalgamated two distinct models based on DenseNet-121 and MobileNet-V2 architectures, incorporating custom layers for training on the teacher's dataset. Without the teacher-student model, accuracy was 93.87%. The collaborative efforts of the teacher and student ensemble yielded an accuracy rate of 98.47% utilising parameters 5,91,108, indicating the model's lightweight characteristics.

The rest of the paper is organised in the following manner. Section 2 discusses the related works. The basic techniques of pre-trained models are introduced in Section 3. The proposed methodology is presented in Section 4. Section 5 provides experimental results. Finally, the paper concludes with Section 6.

II. RELATED WORK

Paddy Doctor [13] proposed a study using Convolutional Neural Networks (CNN) and transfer learning: VGG16, MobileNet, Xception, and ResNet34. The used dataset consists of 16,225 images of paddy leaves divided into 12 classes for diseases and one for healthy leaves. According to the experimental results, ResNet34 had the highest F1-score of 97.50%. Shweta Lamba [14] proposed a GCL model for

disease classification. GCL uses CNN for feature extraction, LSTM for classification, and GAN for data augmentation. The model achieved 97% accuracy on a dataset from GitHub, UCI, Mendeley, and Kaggle. MS-DNet [15] provides a lightweight network design for effective plant disease detection. The model scored 98.32% accuracy, outperforming any current model. It combined publicly accessible data (the PlantVillage dataset) with locally gathered images of paddy and corn plant diseases. Shankaranarayanan Nalini [16] proposes a DNN classification model for paddy leaf disease detection. The model achieved 96.96% accuracy, 95.92% precision, and 96.41% recall during testing. This suggests that DNN CSA has potential for real-time plant disease detection on devices. This literature review strengthens the foundation for this research on paddy leaf disease detection using transfer learning. Studies like Paddy Doctor [13] demonstrate the effectiveness of transfer learning with CNNs, achieving high accuracy (97.50% F1-score). This validates this research approach and provides a benchmark for success. Nalini [16] demonstrated the high efficiency of Convolutional Neural Networks in detecting rice diseases. Additionally, Lamba [14] highlights the value of pre-trained models, supporting this research utilisation of YOLOv8's pre-trained CNN capabilities. Overall, these findings firmly support the chosen research method.

Md. Saiful Niaz [17] proposed a model based on YOLOv8 and CNN models such as Inception-Resnet-V2, ResNet-101, and Xception. The dataset consists of 1224 images to train the model. Rectified Linear Unit (ReLU) activation function and Max pooling were used to represent the complex association with the dataset and to reduce the feature maps' spatial dimensions. The model achieved 99.16% accuracy. The paper of Chinna Gopi Simhadri [18] focused on the application of transfer learning and deep learning techniques to automatically detect rice leaf diseases with 15 pre-trained CNN models. The InceptionV3 model outperformed other models with an average accuracy of 99.64%, where a dataset of 10,080 images of ten rice leaf diseases is used. R. P. Narmadha [19] suggested DenseNet169-MLP for rice plant disease detection with an accuracy of 97.68% that utilises DenseNet169 as a feature extractor. Vinay Gautam's [20] proposed model using InceptionV3, VGG16, ResNet, SqueezeNet, and VGG19 achieved an accuracy rate of 96.4% along with a precision of 66.3% for identifying diseases with 1500 leaf images taken from Kaggle and Mendeley. VGG-19, Inception-ResNet-V2, ResNet-101, and Xception have been used for paddy leaf disease detection by Md Ashiqul Islam [21] and Inception-ResNet-V2 has achieved the highest accuracy of 92.68% among them. The suggested solution offers an efficient method to develop a high-quality object detector. We

have used transfer learning instead of starting from scratch to train YOLOv8, which is a powerful object identification model. Pre-trained CNNs are already highly excellent at differentiating basic forms and textures in photos. Thus, training time is significantly decreased by applying the information to the original layers of YOLOv8 from these pre-trained CNNs.

Ali Ghofrani [22] provides an approach for increasing the accuracy of a small model by utilizing information from a larger model. The PlantVillage dataset consists of 54,306 photos of healthy and sick plant leaves. The study achieves 97.58% accuracy on MobileNet, which is close to a large Xception model on the server's 99.73% accuracy. The small model's classification rate is increased by 2.12% employing the teacher-student approach. The proposed method of Yanxin Hu [23] uses novel additions to the YOLOv5s model and uses channel-wise knowledge distillation in model training for boosting the detection of maize leaf diseases with a dataset of 12,957 images containing five leaf diseases. The suggested approach contains 15.5% fewer parameters than YOLOv5s but achieves higher mAP(0.5) and mAP(0.5:0.95) scores of 3.8% and 1.5%, respectively. A low-power deep learning model to identify plant disease in smart hydroponics is proposed by Aminu Musa [24] with knowledge distillation techniques which contains a simple network structure with reduced parameters that results in lower computational requirements and power consumption. An accuracy of 99.4% is achieved by this model with a maximum power consumption of 6.22W, which is significantly lower compared to existing models. In terms of accuracy and power consumption, the proposed low-power student model outperforms other models.

Below Table I gives a gist of the related works that used KD in their core work compared to our work.

Table I
COMPARISON OF TECHNIQUES
USED IN RELATED WORKS (KD).

Ref.	Efficiency	Relevance	Overfitting	Innovation
Musa et al. [11]	Low powered	-	-	-
Hu et al. [12]	Less parameters	-	Not mentioned	Standard KD
Ali Ghofrani [22]	Accuracy +2.12%	-	-	-
Our work	Achieved 97.65% accuracy, <50% parameters, KD + QAT + PQT.	Limited resources	Early Stopping, Data Augmentation, batch dataset training	Combo of KD, QAT, PQT

III. PRE-TRAINED MODELS

Pre-trained models are machine learning models that have been trained on large datasets for specific tasks, such as recognising objects in images or understanding text. They come ready-made with learned features, which can save time and resources when building new models. By leveraging these pre-trained features, developers can create more accurate and efficient models for their own tasks without starting from scratch. Pre-trained models have become popular due to their ability to provide state-of-the-art performance and save development time. The pre-trained models we used in this paper are stated in Table II.

Table II
PRE-TRAINED MODELS USED TO TEST DATASET
PERFORMANCE. T-PARAMETERS = TRAINABLE PARAMETERS

Models	T-Parameters	Dimension	Batch	Acc
Densenet-121	10,54,725	-	-	96.93%
Mobilenet-V2	13,16,869	-	-	94.38%
Inception-V3	21,01,251	-	-	91.33%
Xception	229,09,228	224x224	32	89.67%
VGG-19	1236,47,861	-	-	87.63%
NasNetMobile	15,85,897	-	-	87.24%
Googlenet	27,48,981	-	-	72.44%

These pre-trained models are used to find the best teacher among them. They have been tested with 3 datasets [25] [26] [27] and used "ImageNet" weights for training the classifiers. We used an early stop with "Patience 5" to tackle overfitting of the models. Therefore, these results show the best values from the training, these values may differ without early stopping. From the scenario, it is visible that DenseNet-121 performed the best with faster data accuracy. Therefore, we used "DenseNet-121" as our teacher model.

IV. METHODOLOGY

Using Knowledge Distillation (KD) alone in the Teacher-Student Model was our first approach. However, we discovered that we had to use Quantization Aware Training (QAT) and Post Training Quantization (PTQ) because we had chosen to use both pre-trained Models in the Teacher and Student roles. This choice was chosen since KD by itself cannot shrink the pre-trained Model's size because its parameters are fixed. To guarantee that the Student Model might be successfully quantized in the future, we applied QAT to both the Teacher and Student Models. By including quantized weights, bias, activation, and convolutional layers in the Model, QAT emulates the effects of quantization. A key component of the network that adds flexibility to the learning capabilities of CNNs is the activation function that allows for non-linear processing of data from the input. Whereas, Bias is an additional parameter in neural networks, including CNNs, that allows the model to fit the

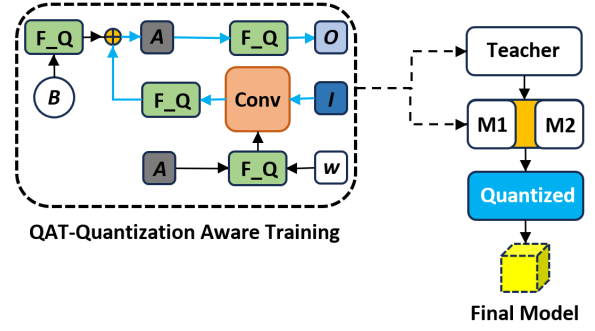


Figure 1. Quantization Applied to both Teacher and Student Model. F_Q = Fake Quantization, W = Weights, A = Activation Function, B = Bias, O = Output, I = Input

data better by shifting the activation function. Bias helps the model learn more flexible patterns in the data. The way that QAT replicates the effects of quantization is seen in the diagram below Figure 1. M1 and M2 denotes two Student Models.

In order to develop a composite Student Model, our study combines the QAT-DenseNet-121 and QAT-MobileNet-V2 Models. This approach leverages the unique strengths of both architectures: DenseNet-121 excels at complex feature extraction, while MobileNet-V2 excels at high-level feature abstraction. Combining their separate outputs gives the Model a wider range of features to choose from, which improves overall generalization and reduces over-fitting.

These Models are comprised of Convolution layers, Translation layers, Dense blocks, and Linear Bottlenecks. Convolutional layers are composed of a 7x7 conv2D operation with a stride of 2, Batch Normalization for mitigating internal co-variate shift, ReLu activation function, and Max Pooling of size 3x3 with a stride of 2. The Dense Blocks are comprised of numerous convolutional layers with interconnecting connections between each adjacent layer. Translation layers increase the ability to manipulate feature mapping through the process of down sampling. The utilization of Inverted Residual Blocks in this context involves the integration of depth-wise convolution and point-wise convolution, which are then linked with a Linear Bottleneck structure. This bottleneck increases the spatial resolution of the feature maps before downsizing it, allowing for the extraction of intricate patterns. Contrastingly, Residual Connection serves to enhance the propagation of gradients during the training process. Finally, we extracted the results of these two algorithms through the process of Global Average Pooling. The ultimate results of these individual minimalist structures are consolidated into a 256 Dense layer tailored to the specific task at hand, and are aggregated for a four-class categorization with subsequent output forecasts all are drawn in Figure 2.

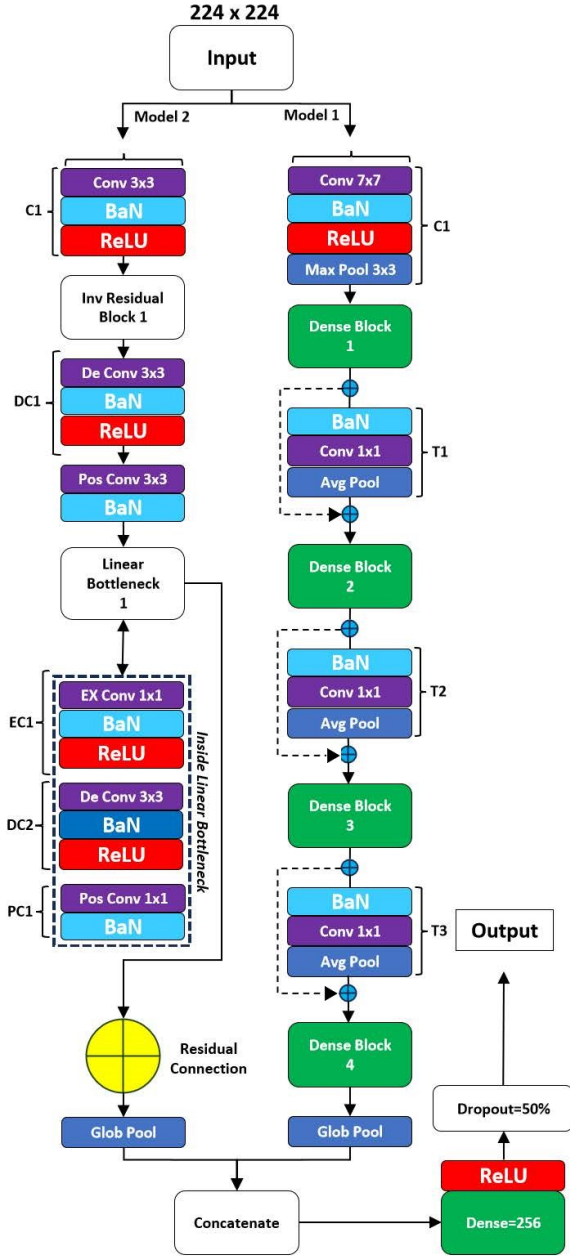


Figure 2. The proposed model architecture.

The Student neural network is trained in conjunction with the expertly trained Teacher neural network (KD). In this scenario, forward propagation is executed on both the mentor and learner networks, followed by back-propagation solely on the learner network. Two loss functions Figure 3 have been specified. One factor to consider is the loss in Student and the loss in distillation function. In addressing Student Loss, the Categorical cross-entropy function is employed, while the Kullback-Leibler Divergence (KL-Divergence) loss function is utilized for Distillation Loss. The Kullback-Leibler Divergence loss function is utilized

for determining the dissimilarity between two probability distributions. The objective is to reduce the Kullback-Leibler Divergence in order to align the predictions of the Student network with those of the Teacher network.

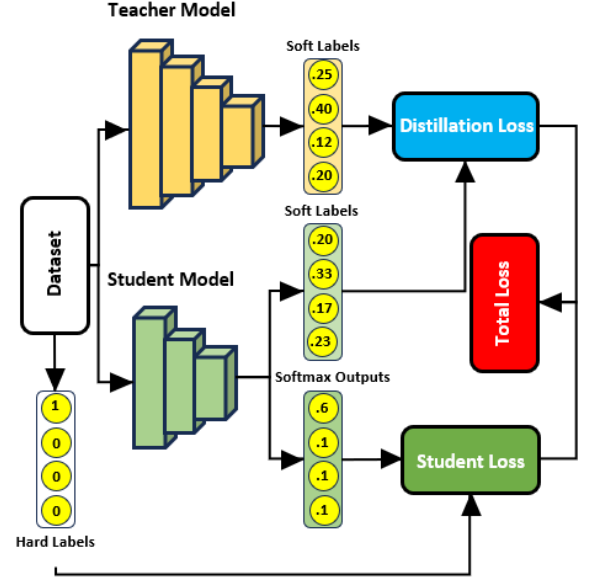


Figure 3. Knowledge Distillation Architecture.

Hard labels in machine learning use binary or one-hot encoding to assign a single, definitive class to every data point. They are a typical representation of assurance in supervised learning tasks. Conversely, Soft labels show a probability distribution among classes, signifying ambiguity and offering more details regarding the Model's confidence in certain results. Soft labels are especially helpful in processes such as KD Figure 3, where they facilitate the learning of a smaller Model (Student) from the richer outputs of a bigger Model (Teacher), hence enhancing performance and generalization.

The Softmax with temperature function adjusts the confidence of the Softmax output (Soft Labels) by introducing a temperature parameter T . This controls the sharpness of the probability distribution. When $T = 1$, the function behaves like the standard Softmax. For $T > 1$, the output probabilities become more spread out and less confident, while for $T < 1$, the probabilities become sharper, focusing more on the largest logit values. The equation is given as:

$$(SoftLabels)\sigma(z_i) = \frac{\exp\left(\frac{z_i}{T}\right)}{\sum_{j=1}^n \exp\left(\frac{z_j}{T}\right)} \quad (1)$$

where z_i is the i -th logit, T is the temperature, and n is the number of classes.

$$(DTL)L_{KL}(p^s(\tau), p^t(\tau)) = \tau^2 \sum_j p_j^t(\tau) \log \frac{p_j^t(\tau)}{p_j^s(\tau)} \quad (2)$$

KL Divergence used in Distillation Loss (DTL), denoted as L_{KL} in equation (2), measures how different the Student's softened probability distribution is from the Teacher's. The goal of distillation is to minimize this divergence, meaning the Student Model adjusts its output probabilities to closely resemble the Teacher's. The equation sums over all classes, with the Teacher's probability for each class $p_j^t(\tau)$ guiding the log ratio between the Teacher's and Student's probabilities. This way, the Student is encouraged to mimic the Teacher's output distribution.

Additionally, the τ^2 scaling factor ensures that the temperature appropriately influences the loss. When temperature is high, it places more importance on aligning the relative probabilities across all classes, rather than just focusing on the top prediction.

Student Loss in KD uses the Cross-Entropy loss functions also known as "hard target loss" because it directly compares the Student Model's output with the true labels. This loss is typically combined with the distillation loss (which encourages the Student Model to mimic the Teacher's behavior). The equation goes like this:

$$(StudentLoss) \mathcal{L}_{CE}(\mathbf{p}^s(1), \mathbf{y}) = \sum_j -y_j \log p_j^s(1) \quad (3)$$

The formula shown is the categorical cross-entropy loss, denoted as $\mathcal{L}_{CE}$. This type of loss is commonly used for classification tasks where $\mathbf{y}$ represents the true label (one-hot encoded) and $\mathbf{p}^s(1)$ represents the predicted probability distribution from the Student Model. In this equation: y_j is the true label for class j , $p_j^s(1)$ is the probability assigned by the Student Model to class j . The summation is over all classes j . The loss measures the difference between the predicted probabilities $\mathbf{p}^s(1)$ and the true distribution $\mathbf{y}$. The goal is to minimize this loss during training, which encourages the Student Model to produce a probability distribution that closely matches the true labels.

Finally the total loss $\mathcal{L}$ is a weighted sum of the categorical cross-entropy (CE) loss between the Student's predictions and the true labels, and the Kullback-Leibler (KL) divergence between the softened outputs of the Student and Teacher Models.

The equation is:

$$(TotalLoss) \mathcal{L} = (1-\alpha) \mathcal{L}_{CE}(\mathbf{p}^s(1), \mathbf{y}) + \alpha \mathcal{L}_{KL}(\mathbf{p}^s(\tau), \mathbf{p}^t(\tau)) \quad (4)$$

In this equation: $\mathcal{L}_{CE}(\mathbf{p}^s(1), \mathbf{y})$ is the standard cross-entropy loss between the Student's predictions $\mathbf{p}^s(1)$ (for a temperature of 1) and the true labels $\mathbf{y}$. $\mathcal{L}_{KL}(\mathbf{p}^s(\tau), \mathbf{p}^t(\tau))$ is the Kullback-Leibler divergence between the softened

output probabilities of the Student Model $\mathbf{p}^s(\tau)$ and the Teacher Model $\mathbf{p}^t(\tau)$, where τ represents the temperature applied to smooth the probabilities. α is a hyperparameter that balances the contribution of the cross-entropy loss and the KL divergence. When α is closer to 1, more weight is given to matching the Teacher's softened predictions, and when it is closer to 0, more weight is given to fitting the true labels.

This combination of losses enables the Student Model to learn both from the ground-truth labels and the generalization captured by the Teacher.

V. EXPERIMENTAL ANALYSIS

1. Dataset Description

In this paper, we have selected three diseases and one healthy leaf. Those diseases are brown spot, leaf blast, leaf blight. The dataset we have used is a combination of three different datasets Dataset-1, Dataset-2, Dataset-3 are publicly available datasets collected from Kaggle and Roboflow. Table III shows some detail about the diseases.

Table III
SYMPTOMS OF PADDY DISEASES [17]

Type	Spot Shape	Spot Color	Image
Brown Spot	Circular or Pinhead shaped	Yellow brown	
Leaf Blast	Elliptical	Gray, white centers, brown to red brown margins	
Bacterial Leaf Blight	Stripes	Grayish green and curl up	

Total images are 3774 among them training, validation and testing set is split into 80%, 15%, 5% accordingly. We have used data augmentations in the original datasets to increase the variation for the Model which includes shear_range=0.2, zoom_range=0.2, horizontal_flip=True, rotation_range=20, width shift range=0.2, height shift range=0.2, vertical_flip=True. So, the total images (1.928 x 3008) that the Model will be trained on will be 5799 images.

Allocating 80% of the data for training, 15% for validation, and 5% for testing enhances model development by supporting effective tuning and assessment. A larger validation set enables reliable hyperparameter optimization, model selection, and early stopping by providing a more accurate representation of unseen data, thus reducing the risk of overfitting or underfitting. It also ensures stable performance metrics, making it easier to gauge the model's

progress during training. By keeping the test set minimal and untouched, it remains an unbiased measure of final model performance, ensuring that the test results accurately reflect generalization to truly new data.

2. Experimental Setup and Results

The training device is equipped with an AMD Ryzen 7 4800H processor, 16 GB of RAM, and a dedicated NVIDIA GeForce 1650 GPU with 4 GB of VRAM. In order to compile the code, the team employed Jupyter Notebook and the TensorFlow libraries. TensorBoard was employed for visualising the data. We have leveraged DenseNet-121 as the reference Model for our knowledge distillation approach, transferring its learned weights and parameters to our Student Model. The training process began by running each model for 100 epochs, repeated five times, to select the best-performing Teacher model. The chosen Teacher model was then fine-tuned using Quantization-Aware Training (QAT), with three runs of 100 epochs each. In this scenario, we employed a parameter value of $\alpha = 0.6$ and a temperature value of $T = 6$. Finally, Knowledge Distillation (KD) was applied to transfer knowledge from the Teacher to the Student model, combining KD with QAT over four separate runs of 100 epochs.

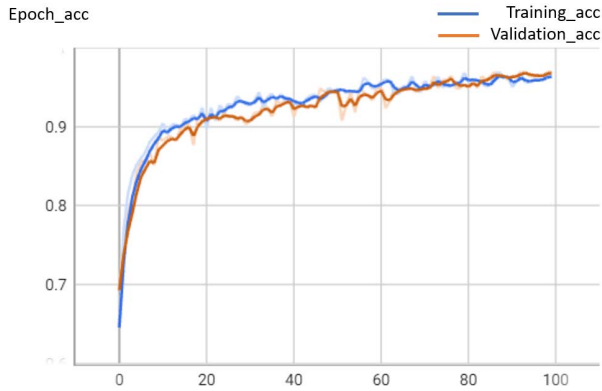


Figure 4. Distilled model training loss vs validation loss.

In the following figures Figure 4 shows how Distillation helped the Student Model to balance between Training Loss and Validation Loss with minimal over-fitting. Figure 5 depicts the Student Loss in various stages as the training carried on. Finally, Figure 6 shows in per epoch the Distillation loss got smoothed out as Teacher helped the Student to learn features more accurately.

When analyzing the Teacher Model in contrast to the Student Model, the Teacher Model had 10,54,725 trainable parameters and Student Model had 5,91,108. With a difference of 4,63,617 instances, the Student Model outperformed the Teacher Model, Table IV shows the performance of the

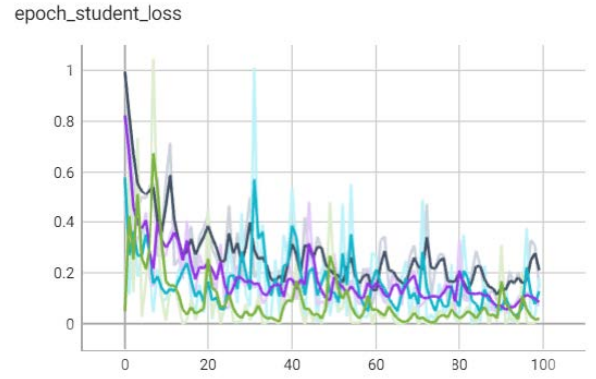


Figure 5. Student loss in multiple tuning stages.



Figure 6. Distillation Loss Per Epoch.

Student Model in multiple performance sections other than accuracy. These results came out to be the best with the combination of parameters we used and will vary depending on different parameter values and machine. Afterwards applying Full Quantization Table V reduced the size of the Student Model.

VI. CONCLUSION

In the course of our research, we successfully achieved identification of three types of diseases in paddy crops: Brown Spot, Leaf Blast, and Leaf Blight. The DenseNet-121 Model achieved the highest accuracy of 96.93%. Afterwards the bespoke dual channel Student Model successfully executed the learning task and surpassed the Teacher's threshold level with a 98.47% accuracy rate. The Student Model achieved increased accuracy with nearly 50% less parameters being optimized. This will facilitate expedited inference, compact footprint, and advantageous for devices with limited memory capacity as Quantization is applied afterwards though the accuracy is also significant 97.65%.

Table IV
DISTILLER MODEL CLASSIFICATION REPORT.

Labels	Precision	Recall	F1 Score
Brown Spot	1.00	1.00	1.00
Healthy	0.92	0.97	0.95
Leaf Blast	0.97	0.91	0.94
Leaf Blight	1.00	1.00	1.00

Table V
TRAINED MODEL ACCURACY REPORT.

Trained Models	Accuracy	Size
Teacher	96.93%	40.0 MB
Student	93.87%	38.7 MB
Student with KD	98.47%	38.7 MB
Student with KD+QAT+PQT	97.65%	18.6 MB

Future research endeavors in this area will involve the utilization of diverse instructional paradigms to educate Student Models across various proficiency levels.

REFERENCES

- [1] S. Sen, R. Chakraborty, and P. Kalita, "Rice-not just a staple food: A comprehensive review on its phytochemicals and therapeutic potential," *Trends in Food Science & Technology*, vol. 97, pp. 265–285, 2020.
- [2] "What is the importance of rice in bangladesh? kernel description," accessed: 2024-03-17. [Online]. Available: <https://typeset.io/questions/what-is-the-importance-of-rice-in-bangladesh-4ibr6r1us>
- [3] R. Reinke, S. Kim, and B. Kim, "Developing japonica rice introgression lines with multiple resistance genes for brown planthopper, bacterial blight, rice blast, and rice stripe virus using molecular breeding," *Molecular Genetics and Genomics*, vol. 293, no. 6, pp. 1565–1575, 2018.
- [4] M. Agrawal and S. Agrawal, "Rice plant diseases detection & classification using deep learning models: A systematic review," *Journal of Critical Reviews*, vol. 7, no. 11, pp. 4376–4390, 2020.
- [5] J. Chen, D. Zhang, Y. Nanekaran, and D. Li, "Detection of rice plant diseases based on deep transfer learning," *Journal of the Science of Food and Agriculture*, vol. 100, no. 7, pp. 3246–3256, 2020.
- [6] S. Jadhav, V. Udupi, and S. Patil, "Identification of plant diseases using convolutional neural networks," *International Journal of Information Technology*, vol. 13, no. 6, pp. 2461–2470, 2021.
- [7] T. Daniya and S. Vigneshwari, "Deep neural network for disease detection in rice plant using the texture and deep features," *The Computer Journal*, vol. 65, no. 7, pp. 1812–1825, 2022.
- [8] J. Li, X. Zhao, H. Xu, L. Zhang, B. Xie, J. Yan, L. Zhang, D. Fan, and L. Li, "An interpretable high-accuracy method for rice disease detection based on multisource data and transfer learning," *Plants*, vol. 12, no. 18, p. 3273, 2023.
- [9] A. Sharma, A. Jain, P. Gupta, and V. Chowdary, "Machine learning applications for precision agriculture: A comprehensive review," *IEEE Access*, vol. 9, pp. 4843–4873, 2020.
- [10] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015. [Online]. Available: <https://arxiv.org/abs/1503.02531>
- [11] A. Musa, M. Hassan, M. Hamada, and F. Aliyu, "Low-power deep learning model for plant disease detection for smart-hydroponics using knowledge distillation techniques," *Journal of Low Power Electronics and Applications*, vol. 12, no. 2, p. 24, 2022.
- [12] Y. Hu, G. Liu, Z. Chen, J. Liu, and J. Guo, "Lightweight one-stage maize leaf disease detection model with knowledge distillation," *Agriculture*, vol. 13, no. 9, p. 1664, 2023.
- [13] B. Petchiammal, Kiruba and P. Murugan, Arjunan, "Paddy doctor: A visual image dataset for automated paddy disease classification and benchmarking," in *CODS-COMAD '23*. Association for Computing Machinery, New York, NY, USA, 2023, pp. 203–207.
- [14] S. Lamba, A. Baliyan, and V. Kukreja, "A novel gcl hybrid classification model for paddy diseases," *International Journal of Information Technology*, 2022.
- [15] W. Chen, J. Chen, R. Duan, Y. Fang, Q. Ruan, and D. Zhang, "Ms-dnet: A mobile neural network for plant disease identification," *Computers and Electronics in Agriculture*, vol. 199, p. 107175, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0168169922004926>
- [16] S. Nalini, N. Krishnaraj, J. Thangaiyan, K. Vinothkumar, A. Britto, K. Subramaniam, and C. Bharatiraja, "Paddy leaf disease detection using an optimized deep neural network," *Computers, Materials & Continua*, vol. 680, pp. 1117–1128, 2021.
- [17] M. Niaz, L. Tarin, T. Shabonty, M. Saad, and D. Farid, "Paddy leaf disease detection employing transfer learning," in *2023 26th International Conference on Computer and Information Technology (ICCIT)*, 2023, pp. 1–6.
- [18] C. Simhadri and H. Kondaveeti, "Automatic recognition of rice leaf diseases using transfer learning," *Agronomy*, vol. 13, no. 4, 2023. [Online]. Available: <https://www.mdpi.com/2073-4395/13/4/961>
- [19] R. Narmadha, N. Sengottaiyan, and R.J.K., "Deep transfer learning based rice plant disease detection model," *Intelligent Automation & Soft Computing*, vol. 31, no. 2, pp. 1257–1271, 2022. [Online]. Available: <http://www.techscience.com/iasc/v31n2/44553>
- [20] V. Gautam, N. Trivedi, A. Singh, H. Mohamed, I. Noya, P. Kaur, and N. Goyal, "A transfer learning-based artificial intelligence model for leaf disease assessment," *Sustainability*, vol. 14, no. 20, 2022. [Online]. Available: <https://www.mdpi.com/2071-1050/14/20/13610>
- [21] M. Islam, M. Shuvo, M. Shamsoddaman, S. Hasan, M. Hos-sain, and T. Khatun, "An automated convolutional neural network based approach for paddy leaf disease detection," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 1, 2021.
- [22] A. Ghofrani and R. Mahdian Toroghi, "Knowledge distillation in plant disease recognition," *Neural Computing and Applications*, vol. 34, no. 17, pp. 14 287–14 296, 2022. [Online]. Available: <https://doi.org/10.1007/s00521-021-06882-y>
- [23] Y. Hu, G. Liu, Z. Chen, J. Liu, and J. Guo, "Lightweight one-stage maize leaf disease detection model with knowledge distillation," *Agriculture*, vol. 13, no. 9, 2023. [Online]. Available: <https://www.mdpi.com/2077-0472/13/9/1664>
- [24] A. Musa, M. Hassan, M. Hamada, and F. Aliyu, "Low-power deep learning model for plant disease detection for smart-hydroponics using knowledge distillation techniques," *Journal of Low Power Electronics and Applications*, vol. 12, no. 2, 2022. [Online]. Available: <https://www.mdpi.com/2079-9268/12/2/24>
- [25] "Rice leaf disease detection obj dataset," visited on 2024-03-18. [Online]. Available: <https://universe.roboflow.com/project-khcjh/rice-leaf-disease-detection-obj>
- [26] "Rice leaf diseases dataset kernel description," accessed:

2023-08-14. [Online]. Available: <https://www.kaggle.com/datasets/vbookshelf/rice-leaf-diseases>

- [27] "Rice diseases image dataset," accessed: 2023-08-14. [Online]. Available: <https://www.kaggle.com/datasets/minhhuy2810/rice-diseases-imagedataset/data>



Md. Saiful Niaz holds a Bachelor of Science in Computer Science and Engineering from United International University. His academic and research pursuits focus on machine learning, deep learning, computer vision, and natural language processing, fields in which he has contributed to several peer-reviewed conference publications. Mr.

Niaz is also credentialed in networking and cybersecurity through Cisco's CCNA program, underscoring his technical proficiency across programming, web development, and robotics. His background reflects a commitment to advancing knowledge in applied computing and intelligent systems.

Email: mniaz201152@bscse.uiu.ac.bd



Tasmia Hassan Shabonty is currently completing her BSc in Computer Science at United International University, Bangladesh. She was awarded the Study in Canada Scholarship 2022 and spent a semester at Lakehead University, Thunder Bay, Canada. Her research interests span artificial intelligence, machine learning, deep

learning, and human-computer interaction (HCI). She is particularly focused on leveraging AI and ML techniques to enhance user interactions and learning experiences in digital platforms. In her recent work, she explores how advanced AI models can be applied within HCI frameworks to create more intuitive and adaptive systems that support user needs.

Email: tshabonty201099@bscse.uiu.ac.bd



Md. Ashfaqur Rahman Saad is currently completing his BSc in Computer Science at United International University, Bangladesh. His research interests include machine learning, deep learning, artificial intelligence, computer networks, and cybersecurity. Saad is particularly interested in the intersection of AI and cybersecurity,

focusing on how machine learning models can be leveraged to enhance network security and protect against emerging cyber threats. His work aims to develop intelligent systems capable of adaptive threat detection and response to strengthen digital infrastructure.

Email: msaad201075@bscse.uiu.ac.bd



Lucky Talukder Tarin is a Computer Science and Engineering graduate from United International University, is committed to advancing technology through impactful research and innovative solutions. Her recent publications highlight her dedication to creating tools that promote transparency, sustainability, and improved outcomes across

various sectors.

Email: ltarin201070@bscse.uiu.ac.bd



Dewan Md. Farid is a Professor of Computer Science and Engineering at United International University. He is an IEEE Senior Member and Member ACM. Prof. He holds a PhD in Computer Science and Engineering from Jahangirnagar University, Bangladesh in 2012. Part of his PhD research has been done at ERIC Laboratory,

University Lumière Lyon 2, France by Erasmus-Mundus ECW eLink PhD Exchange Program. His PhD was fully funded by Ministry of Science and Information and Communication Technology, Government of the People's Republic of Bangladesh and European Union (EU) eLink project. Prof. Farid has published 142 peer-reviewed scientific articles, including 33 highly esteemed journals like Expert Systems with Applications, IEEE Access, Journal of Theoretical Biology, Journal of Neuroscience Methods, Bioinformatics, Scientific Reports (Nature), Proteins and so on in the field of Machine Learning, Data Mining and Big Data.

Email: dewanfarid@cse.uiu.ac.bd



Huu-Hoa Nguyen received his Engineering Degree in Computer Science from Can Tho University, Vietnam. He earned his MSc in Information Systems from HAN University, the Netherlands, and his PhD in Informatics from Lyon University, France. Dr. Nguyen is currently a senior lecturer at the College of Information and Commu-

nication Technology, Can Tho University, Vietnam. His research interests span a wide range of computer science topics, including artificial intelligence, machine learning, data mining, knowledge management systems, computer networks, and cybersecurity. He has led multiple international research projects, receiving funding from prestigious sources such as the Horizon-2020 European Commission and the Newton Fund.

Email: nhhoa@ctu.edu.vn

A Deep Learning based Approach for Student Attendance Recognition

Minh Nguyen Nguyen^{1,2}, Phuong Lan Phan²

¹ Can Tho University, Can Tho, Vietnam

² Can Tho University Software Center, Can Tho, Vietnam

Correspondence: Minh Nguyen Nguyen, nmnguyet@cusc.ctu.edu.vn, Phuong Lan Phan, pplan@cit.ctu.edu.vn

Received: 14/07/2024, revised: 20/09/2024, accepted: 11/11/2024

Digital Object Identifier:: 10.32913/mic-ict-research-vn.v2024.n3.1314

Abstract: Traditional attendance recognition methods - such as using cards or signing into lists are susceptible to fraud - often consumes considerable time and effort in managing and processing information. With advancing technology and increasing security requirements, modern attendance recognition methods offer higher accuracy and convenience for organizations. This paper proposes an approach utilizing DeepFace technology and the deep learning architectures (MobileNet, ResNet) to face detection, face recognition and face anti-spoofing. The proposed approach is experimented on the existing dataset within the research community (CelebA-spoof) and on a dataset (named CUSC-Student) built by ourselves; and is evaluated by two metrics - Accuracy and F1-Score. From the experimental results, this approach also identifies the suitable model that achieves the high accuracy while requiring less computing resources. Therefore, it can be applied in practice, specifically for recognizing student attendance at our division.

Keywords: *Anti-Spoofing, face detection, face recognition, deep learning*

Tiêu đề: Điểm danh sinh viên theo tiếp cận dựa trên mạng học sâu

Tóm tắt: Các phương pháp nhận diện điểm danh truyền thống - như dùng thẻ hoặc ký vào danh sách - dễ bị gian lận và thường đòi hỏi nhiều thời gian và công sức để quản lý, xử lý dữ liệu. Với sự phát triển của công nghệ và yêu cầu bảo mật ngày càng cao, các phương pháp điểm danh hiện đại mang lại độ chính xác và tiện lợi cao hơn cho các tổ chức. Bài báo này đề xuất một phương pháp sử dụng công nghệ DeepFace kết hợp với các kiến trúc học sâu (MobileNet, ResNet) cho các tác vụ phát hiện khuôn mặt, nhận diện khuôn mặt và chống giả mạo. Phương pháp này được thử nghiệm trên tập dữ liệu sẵn có trong cộng đồng nghiên cứu (CelebA-spoof) và một tập dữ liệu tự xây dựng (được đặt tên là CUSC-Student), được đánh giá bằng hai chỉ số - độ chính xác và F1-Score. Từ kết quả thực nghiệm, phương pháp này xác định được mô hình phù hợp giúp đạt độ chính xác cao, đồng thời yêu cầu ít tài nguyên tính toán hơn. Do đó, phương pháp này có thể ứng dụng vào thực tiễn, đặc biệt trong việc điểm danh sinh viên tại đơn vị của chúng tôi.

Từ khóa: *Anti-Spoofing, face detection, face recognition, deep learning*

I. INTRODUCTION

Automatic and real-time attendance recognition is a necessary activity for any organization, especially for educational institutions, because it helps to manage attendees effectively. Face recognition [1] is an advanced technology with numerous applications in security, surveillance, and user convenience. However, it faces significant challenges, particularly its vulnerability to spoofing attacks. Attackers can use printed photos, videos, or masks to bypass recognition systems, compromising the security and reliability of the technology. With the rise of social media, people, especially the youth, frequently post personal images online. This trend increases the risk of face recognition systems be-

ing spoofed. Criminals can exploit high-resolution images, videos, and masks to deceive these systems, posing serious security risks to institutions.

Deep learning [2] is a machine learning method in the field of artificial intelligence that focuses on building and training deep neural networks to automatically learn and understand data. It is called "deep" because it involves multiple layers of computational units known as neurons. These layers are interconnected, passing data from one layer to the next, thus forming a neural network capable of complex learning. Deep learning models, especially convolution neural networks (CNNs) [3], provide superior accuracy, robustness, and real-time performance in face recognition tasks. These models can handle variations in lighting,

angles, and expressions, ensuring reliable recognition under diverse conditions.

This paper proposes an approach for attendance recognition by combining face detection, face anti-spoofing and face recognition. We then develop a web application for the proposed approach by leveraging existing technologies and methods. In this application, we embed some models, then retrain and fine-tune them to suit our dataset.

The paper is structured into five distinct sections to provide a comprehensive overview of our research. The introduction section outlines the context of our study, explaining the importance of face recognition and anti-spoofing systems and the need for enhanced security measures in automated systems. Following this, the related work section presents the existing studies that have tackled similar challenges, highlights their methodologies and findings. The third section is dedicated to detailing our proposed approach. The fourth section delves into experiments. This section covers the tools, datasets and metrics used for training and evaluating the performance of approach, and presents the experimental results. Finally, the conclusion section summarizes the key findings of our study as well as suggests the work future.

II. RELATED WORK

Recent advancements in machine learning have spurred significant progress in the integration of face recognition with anti-spoofing techniques. This intersection of technologies has been driven by several seminal studies.

The authors of research [4] provide an extensive survey of deep learning methods used for face anti-spoofing, covering both single- and multi-modal techniques. The authors review a variety of deep learning models and their application in preventing spoofing attacks such as those involving photos, videos, and 3D masks. The survey highlights the integration of these models with face recognition systems and discusses their effectiveness in improving security and reliability. The comprehensive analysis offers insights into the strengths and weaknesses of different approaches, guiding future research in this field.

The authors of research [5] introduces a novel approach for face anti-spoofing that leverages patch-based and depth-based Convolutional Neural Networks. The methodology involves dividing the facial image into smaller patches and using a CNN to extract features from these patches. Additionally, a depth-based CNN is employed to estimate the depth map of the face, which helps in distinguishing between real and spoof faces by analyzing the texture and depth information. The experimental setup includes two CNN architectures: one for patch-based feature extraction and another for depth estimation. The patch-based CNN

analyzes local regions of the face, providing detailed texture information, while the depth-based CNN focuses on the overall depth profile of the face. These two models work in tandem to enhance the anti-spoofing performance. The author utilizes publicly available datasets for training and evaluation, including the CASIA Face Anti-Spoofing Dataset and the Replay-Attack Dataset. The proposed method demonstrates significant improvements in detecting spoofing attacks compared to traditional approaches. The combination of patch-based and depth-based CNNs already achieves 89% reduction of Equal Error Rate, with notable performance on challenging datasets.

The authors of research [6] propose a new approach for tackling the face of anti-spoofing challenges. The method focuses on learning polysemantic spoof traces by disentangling facial features across multiple modalities. This approach enhances the detection of spoof attacks by effectively separating genuine facial attributes from spoofed ones using multimodal information. The research contributes to advancing the face recognition security, leveraging innovative machine learning techniques to address complex spoofing threats in practical scenarios.

Besides, related to facial recognition and facial attribute analysis, DeepFace technology [7] is applied in authentication, security systems, social media, etc. DeepFace, developed by Facebook, utilizes deep neural networks to detect and recognize faces with high accuracy. The process begins with image preprocessing, where face detection algorithms (such as Viola-Jones or HOG + SVM) identify the position of the face in the image. The detected face is then aligned to ensure that the features of the face such as eyes, nose, and mouth are in inconsistent positions across all images, reducing variations caused by angles and expressions. The core of DeepFace is its deep neural network architecture, which includes convolution layers to extract essential features like edges, contours, and textures from the face image. These features are then consolidated through fully connected layers, culminating in an embedding layer that converts the features into a fixed-length vector representation (often 4096 dimensions). This vector, or face embedding, effectively represents the face in a feature space. For face recognition, the embedding of a new face is compared with the embedding of known faces in the database using similarity measures like Euclidean distance or cosine similarity. If the distance between the new face embedding and a known face embedding is below a certain threshold, the new face is recognized as matching the known face. Additional verification steps can be performed to ensure high accuracy, especially in security-sensitive applications.

In image classification and object detection, the archi-

structures such as YOLO, GoogLeNet, ResNet, VGG and MobileNet are popular. YOLO [8] is a unified architecture which uses a single convolutional network for classification and bounding box prediction, focusing on real-time object detection. GoogLeNet (or Inception-V1) [9] is a deep convolutional neural network architecture which employs inception modules for multi-scale feature extraction and emphasizes the improved utilization of the computing resources inside the network. ResNet [10] is a deep learning architecture with hundreds or even thousands of layers. It is designed to improve the training of very deep neural networks by incorporating residual blocks and skip connections which enables gradients to flow more easily through the network. VGG [11] is a deep CNN architecture with a simple and uniform design. It uses small convolutional filters throughout the network in capturing fine details in images. MobileNet [12] [13] [14] is a lightweight CNN architecture, focused on efficient deployment on mobile and resource-constrained devices. MobileNetV2 [13] and MobileNetV3 [14] are often considered for developing applications. MobileNetV2 is an improved version of the MobileNet architecture, designed to enhance the performance and efficiency of CNNs on mobile devices. It introduces two key concepts: Inverted Residuals and Linear Bottlenecks. These concepts help preserve more information and reduce computational requirements. MobileNetV2 uses three main layers for each block: 1x1 Convolution with ReLU6 to expand the dimensionality, Depthwise Convolution with ReLU6 to each channel separately, and 1x1 Linear Convolution to compress the dimensionality back. This architecture achieves higher efficiency and performance, making it suitable for mobile and embedded applications. Additionally, it includes hyperparameters to balance speed and accuracy by adjusting the number of channels and input resolution. MobileNetV3 Large and MobileNetV3 Small are two main variants of MobileNetV3, but they have distinct architectures and use cases. MobileNetV3 Large integrates optimization techniques from MobileNetV2 with new ideas from MnasNet. It employs ReLU6 and h-swish (Hard-Swish) activation functions to enhance performance. MobileNetV3 Large delivers high performance while being efficient and resource-saving. Therefore, it is suitable for various applications, including image classification and object detection. MobileNetV3 Small is optimized for efficiency and speed on low resource devices. It has fewer layers and parameters. MobileNetV3 Small also uses the Hard Swish activation function to improve performance, selects Squeeze-and-Excitation modules to enhance feature recalibration, and uses Network Architecture Search to fine-tune the architecture. The comparison results of ResNet, GoogLeNet and VGG models presented in researches [10] [11] [15] [16] have shown that: the test error of VGG to be

competitive with that one of GoogLeNet; the test error of ResNet to be the smallest and its accuracy is the largest. The experimental results in [12] shows that MobileNet is more accurate than GoogLeNet while being less computation (the number of multi-adds) and smaller (the number of parameters). According the research in [13], MobileNet achieves competitive accuracy and requires fewer parameters and computations than YOLO and ResNet. The object detection models based on deep learning have been surveyed by [16] shows that for resource constrained environments, the MobileNet is competitive on classification accuracy, latency, and number of parameters. MobileNet has a lightweight architecture, requires less memory and computing resources whereas it achieves competitive accuracy compared to other architectures. It is the balance between performance and accuracy.

Education plays a crucial role in the development of a nation. The number of educational institutions is increasing due to the need for learning, competition in the labor market, technological development, and educational policies. In educational institutions, automatic and real-time attendance recognition is essential because it supports the learner management fast and effectively. However, educational institutions may face equipment limitations due to financial constraints. As a result, the quality and the availability of devices such as computers and cameras can be affected. Devices may not have strong configurations or have limited computing resources. For the attendance recognition problem, if the computer system is not powerful enough, using ResNet or VGG or YOLO can lead to slow and inefficient processing speed because they require more computing resources. Therefore, MobileNet is the best choice in this case.

III. PROPOSED APPROACH

The proposed approach for attendance recognition based is shown in Figure 1 and Algorithm 1. This proposal consists of 3 stages playing the distinct roles.

The first stage “Face Detection” is used to process images, detect faces and recognize faces. The DeepFace technology is utilized to detect and crop the original image, focusing solely on the face and excluding the distracting details like backgrounds, hands and legs. The “Yunet” algorithm is employed to detect faces in the image, adjust their alignment, and crop them based on the coordinates and dimensions of the faces. Subsequently, these processed images are stored for further processing and analysis in anti-spoofing models.

The second stage “Face Anti-Spoofing” is used to distinguish between real face images and fake face images. This stage embeds four models: MobileNetV2, MobileNetV3

Large, MobileNetV3 Small and ResNet18. Depending on the resource constraints of devices, one of four models will be used. The selected model will identify if the face is fake or real. If the face is determined as real, the extracted real face image will be the input of the next component. Therefore, it helps enhance the accuracy and security of the recognition system.

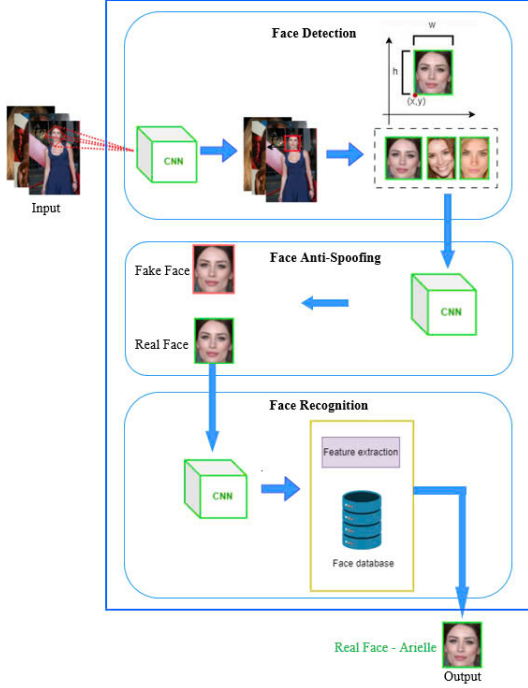


Figure 1. The proposed approach for attendance recognition.

The last stage “Face Recognition” plays a crucial role in converting each face image into a unique feature vector (embedding) by using the ArcFace [17]. This process involves computing embeddings for each image, generating a unique ID for each new face, and adding these embeddings to the FAISS [18] index for efficient management and retrieval. The last stage also utilizes DeepFace to maintain a mapping between the face IDs and usernames in the database; consequently, the process of computing and managing face embeddings becomes efficient, supporting optimal performance for face recognition.

A web application is developed to implement the proposed approach. As shown in Figure 2, this application uses HTML and CSS to design interfaces; JavaScript to send requests; functions written in Python programming language to handle requests from the user interface; and MySQL to store, select and update student information and attendance records.

The application also integrates the following. Leveraging these advanced technologies ensures security and accuracy in recognizing the attendance.

Algorithm 1: attendance_recognition(frame)

```

1 size ← (224, 224);
2 detector ← "yunet";
3 face_objs ← extractFaces (frame,
    size, detector);
4 foreach face_obj in face_objs do
5     confidence ← getConfidence (face_obj);
6     if confidence ≥ 0.9 then
7         facial_area ← getFacialArea (face_obj);
8         model ← selectModel();
9         antispoof_flag ← predictFace (frame, facial_area,
            model);
10        if antispoof_flag == 1 then
11            face ← getFaceFrame (facial_area);
12            presenter ← "Arcface";
13            embeddings ← representFace (face, presenter);
14            indices ← recognizeFaces (embeddings);
15            recognized_name ← getName (indices);
16            displayInformation (recognized_name);
17            updateDatabase (recognized_name);
18        end
19    else
20        showError();
21    end
22 end
23 end

```

- “Yunet” algorithm in DeepFace for image processing and face detection.
- MobileNetV2, MobileNetV3 (Large, Small) and ResNet18 models for face-anti spoofing.
- ArcFace method for face recognition.

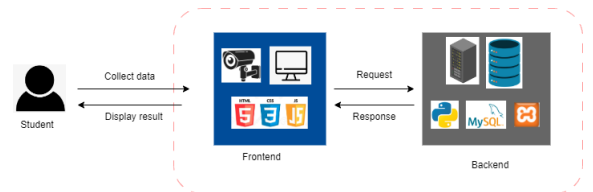


Figure 2. The web application architecture to implement the proposed approach.

The application provides key functions such as user registration and attendance tracking. Initially, students take attendance using a genuine facial image captured by the application. This image undergoes preprocessing using DeepFace technology with the “Yunet” algorithm for image processing and face detection. It is then passed through MobileNetV2, MobileNetV3 (Large and Small) and ResNet18 models for face anti-spoofing to ensure authenticity. Features are obtained from the processed image by the ArcFace method and saved in the embedded database along with the student’s personal information. When a student needs to check in, the application captures their image again, preprocesses it, and runs it through the anti-spoofing models. The extracted features are searched in the database. Attendance information is then recorded accordingly. This provides a secure and efficient method for managing student attendance. The results will be visually displayed on the user interface. If the attendance is successful, the screen will show the student’s photo, a message confirming that the face is real, and the student’s information. Conversely, if no face is detected or if a fake face is identified, the application will present the corresponding alert, as depicted in Figure 3.

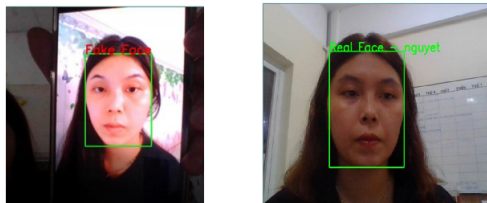


Figure 3. Displaying the result of the user’s face detection.

IV. EXPERIMENTS

The performance of proposed approach is evaluated by using K-folds cross validation method, four metrics (Accuracy, Precision, Recall and F1-Score), and two datasets (CelebA-Spoof and CUSC-Student).

In K-folds method, a dataset is partitioned into K folds (subsets) of equal size. The model is trained and validated K times, then gets the average. Each time, (K-1) folds are used for training, and the rest fold is used for validating the model. This method helps reduce bias and variance, leading to a more reliable model. We conduct three experiments.

- Experiment 1 addresses the question “Which model is the most suitable for integration into a student attendance recognition application?” In this experiment, we chose four models for comparisons: MobileNetV2, MobileNetV3 Large, MobileNetV3 Small and ResNet18. The reason for this selection is as

follows. As mentioned in the related work, three MobileNet models achieve the competitive accuracy while requiring less memory and computing resources. The RestNet model has the smallest test error and the highest accuracy compared to GooLeNet and VGG.

- Experiment 2 explores the question “How does the face detection stage affect the model’s performance?” This experiment compares the performance of the proposed approach both with and without the face detection.
- Experiment 3 examines the question “Which hyperparameter values should we select for integration into a student attendance recognition application?” This experiment changes the values of three crucial hyperparameters (the learning rate, the number of epochs, and the batch size) to identify those that yield optimal performance.

1. Datasets

a) CelebA-Spoof dataset

The CelebA-Spoof dataset [19] is a comprehensive and large-scale resource designed for face anti-spoofing research, featuring 625,537 images from 10,177 subjects. This dataset includes both genuine and spoofed face images, meticulously annotated with 43 detailed attributes related to facial features, illumination, environment, and spoofing techniques. Genuine images are sourced from the CelebA-Spoof dataset, while spoof images are specifically collected and annotated for anti-spoofing purposes. Of the 43 attributes, 40 pertain to genuine images, encompassing a wide range of facial elements and accessories, including skin, nose, eyes, eyebrows, lips, hair, hats, and eyeglasses. The remaining 3 attributes describe spoof images, detailing spoof types, environmental settings, and lighting conditions.

CelebA-Spoof is a critical dataset for training and evaluating algorithms in face anti-spoofing due to its diverse and complex conditions, including various lighting, angles and expressions, which present significant challenges.

However, this proposed approach uses only a subset of the CelebA-Spoof dataset instead of the entire dataset. This decision brings several benefits, such as reducing computational costs and memory usage, while also helping to avoid overfitting. There are 748 folders were randomly selected and each folder contains two sub-datasets: the live folder and the spoof folder. Each folder contains 12,587 real face images and 35,981 fake face images, captured under various conditions, including different colors, face angles, lighting conditions, and backgrounds.

b) CUSC-Student dataset

The CUSC-student dataset is collected and processed by ourselves. We use the web application presented in Section

3 to collect the student photos. The system automatically captures photos when students place their faces in front of the camera. Each student's photo is then saved into individual folders corresponding to each student.

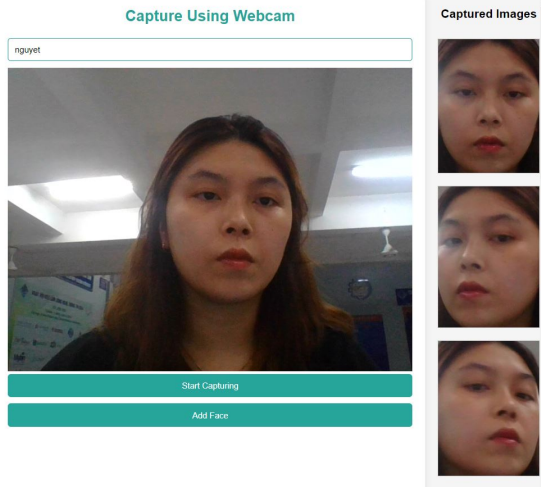


Figure 4. The function “Capture Face” of web application.

The dataset comprises facial images of students participating in short-term or long-term courses at the “Can Tho University Software Center (CUSC)”. The dataset contains 50 folders, each corresponding to a student at CUSC. Within each folder, there are two sub-datasets: one is the live set, containing 4136 real face images of the student, and the other is the spoof set, containing 12,408 fake face images of the student. These images are captured under different lighting conditions and various locations within the CUSC campus, ensuring diverse environmental contexts.

During the data pre-processing phase, images are initially resized to a standardized dimension, ensuring uniform resolution and size across all images. This standardization facilitates streamlined model training processes. The resized images are converted into array format for seamless handling in subsequent preprocessing steps. To further enhance model performance, data augmentation techniques such as scaling, rotation, flipping, and adjustment of lighting conditions are applied. These techniques artificially expand the dataset by creating multiple variations of the initial images.

2. Evaluation Metrics

In face recognition and anti-spoofing, evaluating the model performance is crucial. A popular and effective tool used for this purpose is the confusion matrix[20]. We use a 2x2 confusion matrix, as there are only two classes: real face and fake face. This confusion matrix is presented as follows.

Table I
A 2X2 CONFUSION MATRIX.

	Predicted Real	Predicted Fake
Actual Real	True Positive (TP)	False Negative (FN)
Actual Fake	False Positive (FP)	True Negative (TN)

Where:

- True Positive (TP): The number of instances correctly identified real faces.
- False Negative (FN): The number of instances correctly identified fake faces.
- False Positive (FP): The number of instances incorrectly identified fake faces as real.
- True Negative (TN): The number of instances incorrectly identified real faces as fake.

From the confusion matrix, four metrics - Accuracy, Precision, Recall, F1-Score - are calculated according to formulas (1)(2)(3)(4), respectively. In which, F1-Score and Accuracy metrics are often used together to evaluate the performance of a model.

Accuracy is the proportion of correct predictions out of all predictions. Accuracy measures how often a model correctly predicts the outcome.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

Precision is the proportion of true positive predictions out of all positive predictions. Precision emphasizes correctness.

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

Recall (or Sensitivity or True Positive Rate) is the proportion of true positive predictions out of all actual positives. Recall emphasizes completeness.

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

F1-Score is often used to evaluate the model's performance because it considers both metrics Precision and Recall simultaneously.

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

3. Experimental Results

a) *Experiment 1: Finding the most suitable model for a student attendance recognition application*

In the first experiment, the performance of the proposed approach where one of four models (MobileNetV2, MobileNetV3 Large, MobileNetV3 Small and ResNet18) is

used by the second stage “Face Anti-Spoofing” is evaluated on two datasets: CelebA-Spoof and CUSC-Student. In Experiment 1, these datasets are already processed by the first stage - i.e. it uses the “Yunet” algorithm for face detection and image preprocessing - before training. The goal of this experiment is to assess how each model performs with different datasets to make a decision on the most suitable model for practical deployment.

The experimental results of MobileNet models are presented in two figures (Figure 5 and Figure 6).

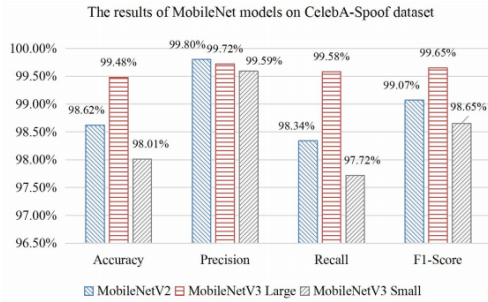


Figure 5. The experimental results of MobileNet models on CelebA-Spoof dataset.

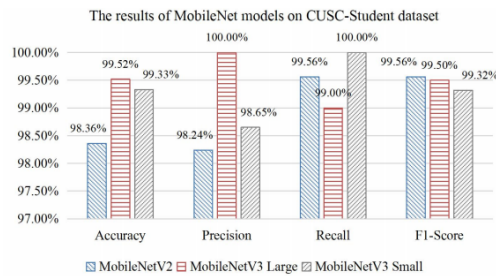


Figure 6. The experimental results of MobileNet models on CUSC-Student dataset.

Among the models trained on the CelebA-Spoof dataset, MobileNetV3 Large shows the best performance, with the highest accuracy at 99.48%, the recall at 99.58%, and the F1-Score at 99.65%. This model effectively balances precision and recall, making it a strong choice for face recognition and anti-spoofing tasks on the CelebA-Spoof dataset. MobileNetV2 also shows good performance, reaching an accuracy of 98.62% and a F1-Score of 99.07%. Although slightly less effective than MobileNetV3 Large, it remains a reliable option. MobileNetV3 Small, while delivering good overall performance, lagged behind the other two models, particularly in recall (97.72%) and F1-Score (98.65%). Despite this, it is still a viable choice, though it may miss more real faces compared to MobileNetV2 and MobileNetV3 Large. Similarly, on the CUSC-Student dataset, MobileNetV3 Large stands out, achieving the highest accuracy at 99.52%, the perfect precision at 100%, and

the F1-Score at 99.50%. This model balances the precision and the recall effectively, making it a dependable option for face recognition and anti-spoofing tasks. MobileNetV2 also demonstrated strong performance, with an accuracy of 98.36% and a F1-Score of 99.56%. It is highly reliable and even surpasses MobileNetV3 Large in recall, indicating its strength in correctly identifying real faces. MobileNetV3 Small, while slightly behind the other two models, still delivered strong performance with the accuracy at 99.33%, the precision at 98.65% and the F1-Score at 99.32%.

We continue to compare the performance of MobileNetV3 Large (the superior model of three MobileNet models) and ResNet18 model. The accuracy and F1-Score of these two models on two datasets are presented in Figure 7. We have also observed the total number of floating-point operations per second (FLOPs) and the total number of parameters (Params) of both models as Table II. The results presented in Figure 7 and Table II shows that MobileNetV3 Large is more remarkable than ResNet18.

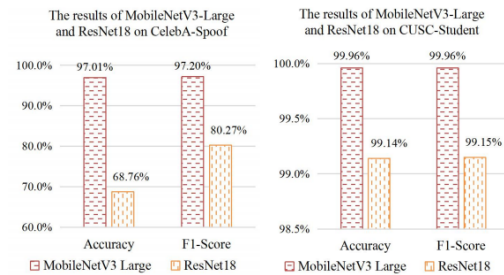


Figure 7. The comparison on the accuracy and the F1-Score of MobileNetV3 Large and ResNet18 models.

Table II
THE COMPARISON ON THE TOTAL PARAMS AND THE TOTAL FLOPS OF MOBILENETV3 LARGE AND RESNET18 MODELS.

Model	Params	FLOPs
MobileNetV3 Large	5.49M	231.32MMac
ResNet18	11.69M	1.83GMac

The experimental results indicate that all models are effective in face recognition and anti-spoofing. MobileNetV3 Large stands out as the better choice for both the CelebA-Spoof dataset and CUSC-Students dataset because it provides high accuracy and the ability to clearly identify the difference between real and fake facial images while requiring less computing resources. Therefore, MobileNetV3 Large is the most suitable model for integration into the student attendance recognition application.

b) Experiment 2: Examining the impact of the face detection stage on the performance of the proposed approach

The aim of this experiment is to compare the performance of the proposed approach with and without the face

detection stage. MobileNetV3 Large model is used in this experiment.

Table III shows the performance of the proposal without the first stage. The input of MobileNetV3 Large remain CelebA-Spoof and CUSC-Student datasets, but images from these datasets are not preprocessed by the first stage.

Table III
THE PERFORMANCE OF THE PROPOSED APPROACH WITHOUT THE FIRST STAGE.

Dataset	Accuracy	Precision	Recall	F1-Score
CelebA-Spoof	89.27%	89.39%	99.76%	93.57%
CUSC-Student	82.53%	77.04%	67.46%	68.42%

From the result of MobileNetV3 Large in Experiment 1 and the collected data in Table IV, the performance of the proposed approach with and without the use of the first stage is shown in Figure 8. The data in Figure 8 indicates that on both datasets, the proposed approach with the first stage will achieve the higher performance.

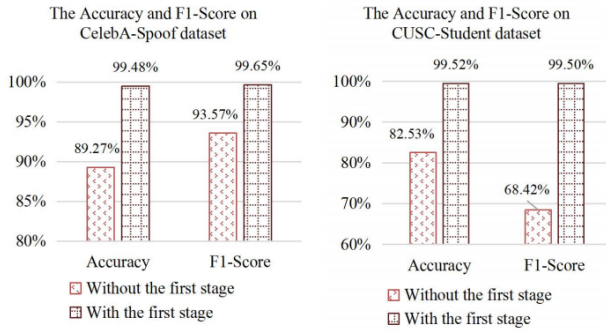


Figure 8. The comparison on the performance of the proposed approach with/without the first stage.

The experimental results indicate that incorporating a face detection phase in the application improves the overall performance of the proposed approach.

c) Experiment 3: Identifying the appropriate hyperparameter values

The MobileNetV3 Large model and its three crucial hyperparameters (the learning rate, the number of epochs, and the batch size) are selected for Experiment 3. The values of these three hyperparameters - a triple set - are changed to evaluate the performance of MobileNetV3-Large on two datasets; thereby identifying the suitable triple set to be used for developing the student attendance recognition application.

The results of Experiment 3 are shown in Table IV where each triple set consists of three values: the learning rate, the batch size, and the number of epochs, respectively.

For the CelebA-Spoof dataset, three different triple sets are tested. The first set (0.005, 128, 20) shows relatively

Table IV
THE PERFORMANCE OF PROPOSED APPROACH WHEN CHANGING THE VALUES OF HYPERPARAMETERS.

Dataset	Triple set	Accuracy	Precision	Recall	F1-Score
CelebA Spoof	(0.005, 128, 20)	68.17%	68.17%	100.00%	81.07%
	(0.002, 64, 30)	78.52%	78.37%	95.29%	83.73%
	(0.0001, 32, 50)	99.48%	99.72%	99.58%	99.65%
CUSC Student	(0.005, 64, 10)	61.57%	56.73%	100.00%	75.73%
	(0.0005, 32, 15)	97.05%	95.56%	100.00%	72.39%
	(0.0001, 32, 20)	95.46%	100.00%	90.60%	95.07%

low performance with an accuracy of 68.17% and a F1-Score of 81.07%. The second set (0.002, 64, 30) performs better, achieving an accuracy of 78.52% and a F1-Score of 83.73%. The third set (0.0001, 32, 50) delivers the best results among 3 sets with an accuracy of 99.48% and a F1-Score of 99.65%. This triple set achieves superior performance regarding accuracy and F1-Score compared to other triple sets.

For the CUSC-Student dataset, the results show that the triple set (0.005, 64, 10) achieves lower performance compared to the other two triple sets. In contrast, the triple set (0.0001, 32, 20) provides the highest performance regarding accuracy and F1-Score, but requires a longer training time. However, the triple set (0.0005, 32, 12) can be considered because it offers a good balance between the model's performance and training time.

The experiment demonstrates that the configuration with a learning rate of 0.0001 and a batch size of 32 consistently performs well across both datasets.

V. CONCLUSION

This paper proposes a face recognition approach combined with anti-spoofing to enhance organizational security. The proposed approach performs well with the MobileNetV3 Large model both on the CelebA-Spoof dataset and CUSC-Student dataset. Real-time applications of this proposed approach can be utilized in various settings, such as schools or universities for student attendance, offices or companies to ensure that only authorized personnel can enter. However, this approach has certain limitations. For example, it may still misidentify when an attacker uses high-quality images, especially as face spoofing and deepfake technologies continue to evolve.

In the future, we will deploy a student attendance system in practice where the networked, high-resolution, motion-tracking cameras will be placed at common locations such

as the classroom entrances. We will then conduct experiments to determine the appropriate distances between the camera and the learner, and the suitable lighting conditions to accurately identify the learner. The proposed approach could be improved to enhance face recognition capabilities in low light, uneven lighting, or continuously changing conditions using adaptive lighting algorithms and image enhancement technologies. Additionally, more complex algorithms could be developed to detect and differentiate between real and fake faces created by deepfake technology. This includes utilizing advanced deep learning models to analyze detailed facial features and behaviors.

REFERENCES

- [1] Almabdy, S., & Elrefaei, "Deep Convolutional Neural Network-Based Approaches for Face Recognition", *Applied Sciences*, 9(20), 4397, 2019. doi:10.3390/app9204397.
- [2] Samira, P., Saad, S., Yilin, Y., Haiman, T., Yudong, T., Maria, P. R., Mei-Ling, S., Shu-Ching, C., & Iyengar, S. S, "A Survey on Deep Learning: Algorithms, Techniques, and Applications", *ACM Computing Surveys*, 51, 1-36, 2018. doi:10.1145/3234150.
- [3] Jones, M., & Lee, S, "An Introduction to Convolutional Neural Networks", *Journal of Artificial Intelligence Research*, 37(2), 45-67, 2021.
- [4] Li, Y., & Liu, Y, " Deep Learning for Face Anti-Spoofing: A Survey", *IEEE Access*, 8, 105383-105404, 2020. doi:10.1109/ACCESS.2020.2997350.
- [5] Xu, X., Li, W., & Zhang, Y, "Face Anti-Spoofing Using Patch and Depth-Based CNNs", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 1-6, 2018. doi:10.1109/CVPRW.2018.00120.
- [6] Li, K., Yang, H., Chen, B., Li, P., Wang, B., & Huang, D, "Learning Polysemantic Spoof Trace: A Multi-Modal Disentanglement Network for Face Anti-Spoofing", *arXiv preprint arXiv:2212.03943*, 2022.
- [7] Taigman, Y., Yang, M., Ranzato, M. A., & Wolf, L. , "DeepFace: Closing the Gap to Human-Level Performance in Face Verification", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1701-1708, 2014. doi:10.1109/CVPR.2014.220.
- [8] Redmon, J., Divvala, S. K., Girshick, R. B., & Farhadi, A, "You Only Look Once: Unified, Real-Time Object Detection", *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 779-788, 2015
- [9] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S. E., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, " Going deeper with convolutions", *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1-9, 2014
- [10] He, K., Zhang, X., Ren, S., & Sun, J. , "Deep Residual Learning for Image Recognition", *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770-778, 2015
- [11] Simonyan, K., & Zisserman, A. , "Very Deep Convolutional Networks for Large-Scale Image Recognition", *CoRR*, 2014.
- [12] Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. , " MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications", *ArXiv*, 2017.
- [13] Sandler, M., Howard, A. G., Zhu, M., Zhmoginov, A., & Chen, L.. " MobileNetV2: Inverted Residuals and Linear Bottlenecks", *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4510-4520, 2018.
- [14] Howard, A. G., Sandler, M., Chu, G., Chen, L., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., Le, Q. V., & Adam, H. , " Searching for MobileNetV3", *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 1314-1324, 2019.
- [15] Ravpreet, K., & Sarbjeet, S. , "A comprehensive review of object detection with deep learning", *Digital Signal Processing*, 132, 103812, 2023.
- [16] Zaidi, S. S., Ansari, M. S., Aslam, A., Kanwal, N., Asghar, M. N., & Lee, B. , " A Survey of Modern Deep Learning Based Object Detection Models", *Digital Signal Processing*, 126, 103514, 2021.
- [17] Deng, J., et al. , "ArcFace: Additive Angular Margin Loss for Deep Face Recognition", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4690-4699, 2019. doi:10.1109/CVPR.2019.00482.
- [18] Douze, M., Guzhva, A., Deng, C., Johnson, J., Szilvasy, G., Mazaré, P. E., & Jégou, H. , " The faiss library", *arXiv preprint arXiv:2401.08281*, 2024.
- [19] *CelebA-Spoof Dataset*. Retrieved from https://mmlab.ie.cuhk.edu.hk/projects/CelebA/CelebA_Spoof.html
- [20] Salmon, B. P., Kleynhans, W., Schwegmann, C. P., & Olivier, J. C. , " Proper Comparison among Methods Using a Confusion Matrix", In *IGARSS 2015*, 2015.



Minh Nguyet Nguyen received with a B.Eng. degree in Data Communication and Computer Networks from Can Tho University, (CTU), Vietnam, in 2022 and is currently pursuing an M.Eng. degree in Information Technology at Can Tho University. Email: nmnguyet@cusc.ctu.edu.vn



Phuong Lan Phan obtained a Master's degree at Asian Institute of Technology, Thailand (2003). She obtained her Ph.D in Computer Science from the University of Science and Technology - The University of Da Nang, Vietnam (2020). Now, she is a lecturer in the Faculty of Software Engineering. Her research interests include

Software Processes, and Recommendation Systems. Email: pplan@cit.ctu.edu.vn