AI-Driven Conversational Interface for Enhancing Industrial Data Accessibility in Warehouse Management Systems

Thaninrath Witoontharaphong[1], **Manussawee Nokkaew**[1†], **Chayada Surawanitkun**[1], **Kanda Sorn-In**[1], **Nongram Mueanrit**[1], **Satit Kravenkit**[2], **Thalerngsak Wiangwiset**[3], and **Apirat Siritaratiwat**[4], Non-members

ABSTRACT

This paper presents an AI-driven conversational interface designed to improve data accessibility in Warehouse Management Systems (WMS). It helps non-technical users access complex warehouse data without SQL knowledge, thus aiding data-driven decisions in manufacturing. Utilizing WangchanBERTa, a Thai language model for Natural Language Processing, the system translates user queries into SQL commands through NL2SQL techniques. The architecture includes five main components and is tested in a footwear manufacturing context. The Intent Classification model achieved a Macro F1-Score of 87.84% and an Accuracy of 94.01%, effectively generating SQL statements for common warehouse queries. However, accuracy drops for more complex intents due to contextual challenges and limited training data. This framework is applicable to other industrial sectors, advancing operational efficiency and human-AI collaboration in Industry 5.0.

Keywords: Natural Language Processing, WangchanBERTa, Warehouse Management System, Artificial Intelligence, Industry 5.0, Smart Factory

1. INTRODUCTION

Industry 4.0 has brought transformative changes to the manufacturing and supply chain sectors by emphasizing system interconnectivity through the Internet of Things (IoT), the implementation of automation, and the generation of Big Data to enhance management efficiency. The integration of various technologies and systems in manufacturing facilities, such as Enterprise Resource Planning (ERP), Manufacturing Execution System (MES), and Warehouse Management System (WMS), serves as the backbone that enables systematic and efficient organizational operations while enhancing competitive capabilities in the digital economy [1].

The transition from Industry 4.0 to Industry 5.0 introduces a distinctly different paradigm that focuses on the close integration of humans with AI-driven technology to create Human-AI Collaboration, enabling more accurate, agile, and natural work processes. While Industry 4.0 emphasized massive data generation, Industry 5.0 focuses on utilizing AI to extract meaningful insights and value from this data for practical applications [2, 3].

WMS represents one of the core systems in modern industrial facilities. Beyond warehouse management capabilities, the system collects and stores comprehensive data on warehouse operations and supply chain activities, including raw material receipt, inventory storage, order picking, packaging, and shipping. This data is crucial for inventory control, error reduction, production planning optimization, and enables management to make accurate strategic decisions [4]. Despite the extensive data collection capabilities of WMS, a significant challenge persists: personnel at various levels, including operational staff, supervisors, and executives, face difficulties accessing required information conveniently. This is primarily due to the complexity of WMS systems, which require technical skills, such as writing SQL queries to retrieve specific information, e.g., "How many pairs of Comfort X shoes size 42 are remaining in stock?" or "When will purchase order PO-2024-001 be shipped?" The inability of users to instantly access such information creates an Accessibility Gap that directly impacts organizational agility and decision-making efficiency [5].

To address this challenge, integrating Natural Language Processing (NLP) and Artificial Intelligence (AI) technologies to develop intelligent interfaces that enable users to query information in natural language without requiring technical commands has gained significant popularity in modern organizations [6, 7]. In particular, the development of Chatbot systems for organizational applications has proven effective, as they can function

Manuscript received on December 23, 2025; revised on January 26, 2026; accepted on July 8, 2026. This paper was recommended by Associate Editor Chanon Warisarn.

[1]The authors are with Center of Multidisciplinary Innovation Network Talent (MINT Center), Faculty of Interdisciplinary Studies, Khon Kaen University, Nong Khai Campus, Nong Khai, Thailand.

[2]The author is with Department of Computer Science, College of Computing, Khon Kaen University, Khon Kaen, Thailand.

[3]The author is with Department of Central Maintenance Sector, National Telecom Public Company Limited, Khon Kaen, Thailand.

[4]The author is with Department of IoT and Information Engineering, Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand.

[†]Corresponding author: manuno@kku.ac.th

©2026 Author(s). This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivs 4.0 License. To view a copy of this license visit: https://creativecommons.org/licenses/by-nc-nd/4.0/.

Digital Object Identifier: 10.37936/ecti-eec.2026243.262941

as "intelligent assistants" that rapidly connect users to organizational databases [8]. The implementation of Chatbot systems in organizations provides several key benefits, including: (1) process simplification by eliminating multi-layer communication between departments, (2) planning support through real-time data access, (3) workload reduction for technical staff by minimizing repetitive queries, (4) cost and time efficiency in communication processes, and (5) enhanced data accuracy through direct database connectivity (detailed in Section 4.4).

However, most existing research has focused on developing Chatbots for English or Chinese languages, targeting general domains such as customer service or general question-answering systems. Meanwhile, the development of Thai language Chatbots for WMS in industrial settings remains limited, especially Natural Language to SQL (NL2SQL) systems that must support domain-specific terminology and complex database structures.

Therefore, this research develops a Thai-language Chatbot to query information from the WMS of the Deblu health footwear manufacturing facility. The primary objective is to create a new channel for accessing information that is simple, convenient, and natural. Users can ask questions in Thai, such as "How many pairs of Comfort X shoes size 42 are in stock?" "Has purchase order PO-2024-001 been shipped from the warehouse?" or "When was leather material for Comfort X model withdrawn for production?" The system analyzes questions using WangchanBERTa, a Thai pre-trained language model with high performance in understanding Thai language context and semantics, for Intent Classification and Entity Extraction [9]. Subsequently, the system converts the processed questions into SQL commands to retrieve data from the WMS database and returns answers to users in an easily understandable and immediate format.

This research presents the architecture, methodology, and experimental results of implementing AI and Thai NLP technologies to enhance data accessibility in WMS environments. The proposed system addresses challenges under the Industry 5.0 paradigm, which emphasizes human-AI collaboration to create human-centered autonomous ecosystems in supply chain operations [10, 11]. This work represents one of the first attempts to develop and deploy a Thai-language NL2SQL chatbot for WMS applications in the footwear manufacturing industry in Thailand. The system not only enhances operational efficiency by enabling natural language queries to complex databases but also provides a replicable framework that can be extended to other industries to improve competitive capabilities through data-driven decision making [12].

2. LITERATURE REVIEW

The development of chatbots for natural-language database querying has attracted significant attention in recent years, particularly in business and industrial domains. Related research can be categorized into four main groups: Chatbot architectures and models, Natural Language to SQL (NL2SQL) systems, Chatbot applications in Supply Chain and WMS, and Thai language question-answering systems.

2.1 Chatbot Architectures and Models

Zhang et al. conducted a comprehensive literature review of business chatbots, examining various model architectures, including Artificial Neural Network (ANN), Recurrent Neural Network (RNN), Convolutional Neural Network (CNN), and Graph Neural Network (GNN), and comparing pipeline and end-to-end structures. The study found that Deep Learning enhances chatbots' ability to understand intent and context, support multi-turn dialogue, and integrate knowledge from external sources. Furthermore, the research proposed integrating with internal organizational systems, such as Customer Relationship Management (CRM), Enterprise Resource Planning (ERP), and Warehouse Management System (WMS). This research demonstrates the importance of designing appropriate architectures for Chatbots in business contexts, particularly for integration with existing organizational systems [13, 14].

2.2 Natural Language to SQL Systems

Converting natural language questions into SQL commands (NL2SQL) has attracted considerable research attention, as it enables non-technical users to access database information. Khadija and Mustapha developed the CHAT-SQL model using a Sequence-to-Sequence (Seq2Seq) architecture with Long Short-Term Memory (LSTM) for converting natural language questions into SQL commands. They used the Spider dataset, a large-scale, complex, cross-domain dataset containing 10,181 questions and 200 databases for training and evaluation. Experimental results from processing 350 questions showed that the model achieved 82.8% Precision, 87.8% Recall, and 89% Accuracy. The model supports both basic SQL commands such as SELECT and WHERE, as well as complex commands including JOIN, GROUP BY, and nested queries. Additionally, it demonstrates multilingual capabilities, showcasing the potential of Natural Language Interface to Database (NLIDB) in improving user-database interactions [15–17].

Similarly, Scholak et al. proposed PICARD, a novel approach for controlling SQL generation from text-to-SQL models by leveraging incremental parsing with constrained decoding to ensure syntactically valid SQL. Experimental results on the Spider dataset demonstrated that using PICARD with T5-based models significantly improves performance, particularly for complex SQL commands. This method reduces the generation of syntactically incorrect SQL, a significant issue with traditional Seq2Seq models [18].

2.3 Chatbot Applications in Supply Chain and WMS

In the domain of supply chain and warehouse management, AI-powered technologies have demonstrated significant benefits. Richey et al. (2023) provided a comprehensive roadmap for AI applications in logistics and supply chain management, emphasizing how Natural Language Processing (NLP) and Machine Learning (ML) can transform traditional operations through intelligent automation and real-time decision support. Their research highlighted that AI-driven systems, including Chatbot interfaces, can effectively address critical challenges in inventory optimization, demand forecasting, and risk management, leading to reduced stockouts and overstock, decreased operational costs, and improved overall supply chain efficiency [19].

Similarly, Verma (2023) explored integrating ChatGPT into supply chain management systems to enhance tracking capabilities and improve query response times. The study demonstrated that conversational AI can serve as an effective interface for accessing complex supply chain data, enabling users to obtain real-time information about shipment status, inventory levels, and delivery schedules through natural language interactions [20].

2.4 Thai Language Question-Answering Systems

For Thai-language question-answering system development, Wongpraomas et al. developed a Thai Question Answering System using Pattern-Matching techniques combined with word embedding to support information retrieval for students, such as class and teaching schedules. The system utilized real data from Khon Kaen University's registration system, Forensic Toxicology course data, and 300 student questions. The system employed cosine similarity to find similarities between user questions and predefined question patterns. Experimental results with 50 real questions showed 82% accuracy. However, the system encountered challenges with sentence-patterning flexibility and with handling synonymous vocabulary, resulting in incorrect answers in some cases. For example, questions "What is Professor X's teaching schedule?" and "When does Professor X teach?" have the same meaning, but the system might fail to match them if sentence patterns differ significantly. The research proposed further development using semantic dictionaries or synonym taggers to address words with similar meanings. The study highlights unique challenges of the Thai language, particularly in managing diverse sentence patterns and synonymous words [21].

2.5 Research Gaps and Contributions of This Work

From the related work review, despite extensive research on Chatbot and NL2SQL system development across various domains and languages, several significant gaps remain. In terms of language and domain, most NL2SQL research focuses on English [15] and utilizes data from general-purpose datasets such as Spider and WikiSQL, which have diverse but non-specific database structures. Meanwhile, NL2SQL development for Thai remains very limited. Existing Thai-language research primarily focuses on retrieval-based or pattern-matching question-answering systems that extract information from documents or simple knowledge bases, rather than on generating SQL queries that can work with complex relational databases [21].

Regarding WMS applications, although research by Richey et al. and Verma demonstrates Chatbot benefits in supply chain and WMS systems, these works focus on using Chatbots for general question-answering, recommendations, or status tracking, without specifically developing NL2SQL systems that can directly query WMS database information. Moreover, they do not support the Thai language, which is the primary language of communication in Thai industrial facilities [19, 20]. Technically, while Thai language models such as WangchanBERTa have been developed and show good performance in classification and NER tasks, they have not been applied to NL2SQL tasks for domain-specific applications like WMS that must support domain-specific terminology such as product codes, product models, purchase order statuses, and complex inter-table database relationships [22].

Therefore, this research aims to address these gaps by proposing the development of a Thai Chatbot for WMS systems. The key contribution of this research is being among the first to apply NL2SQL techniques for the Thai language in industrial domains, leveraging modern Thai language models like WangchanBERTa to handle complex Thai questions and WMS domain-specific terminology, while testing the system with real industrial data to demonstrate practical implementation potential and support the transition to Industry 5.0.

3. THE PROPOSED SYSTEM

This research focuses on developing a chatbot for automated warehouse systems in industrial facilities using Natural Language Processing (NLP) and Machine Learning. The chatbot is designed to understand user queries and provide responses regarding product information within the warehouse inventory. The system workflow is divided into two main phases: the Training Phase and the Deployment Phase. The Training Phase focuses on model construction and learning, while the Deployment Phase handles real-time intent prediction and SQL generation for user interaction. The operational workflow of the chatbot, from receiving user queries to generating appropriate responses, is illustrated in Fig. 1.

3.1 Data Collection

In conducting this research, the researchers collected and generated sample data for model fine-tuning. The dataset comprises three components: user query samples, SQL query examples for creating SQL Query Templates, totaling 2,255 data entries. Sample data entries are presented in Table 1.

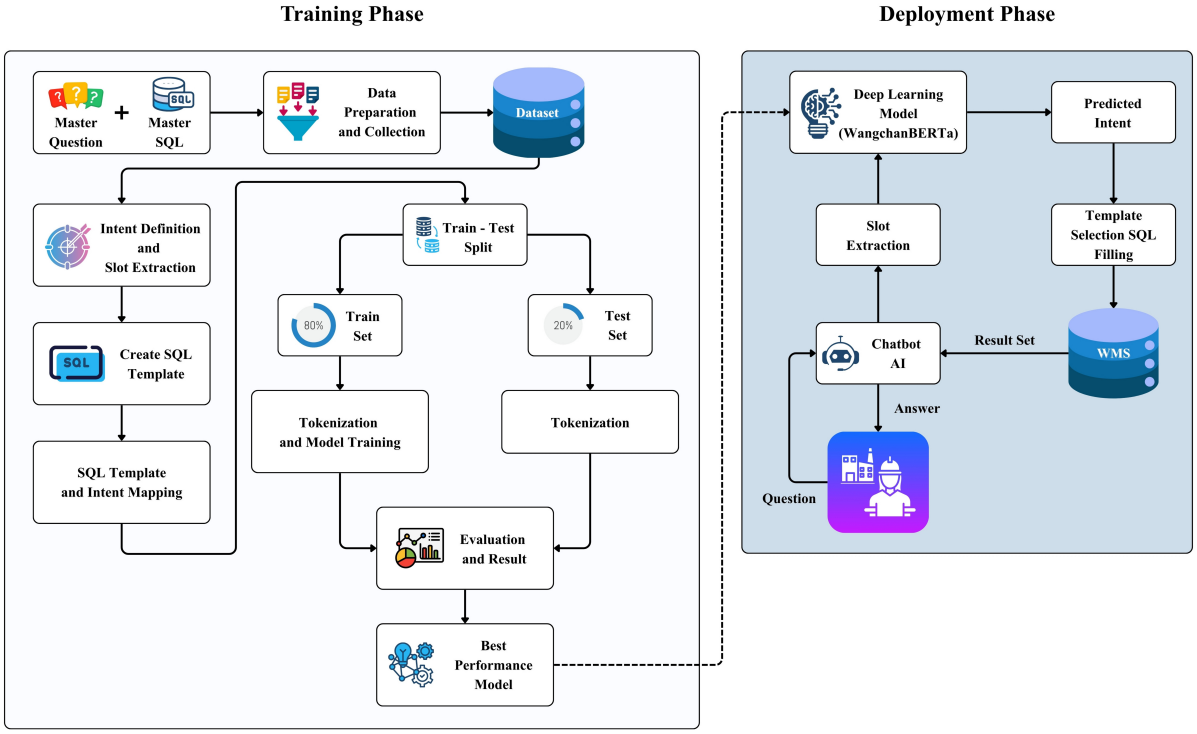


Fig. 1: Overview of the model-training and model-deployment workflow of the proposed Thai NL2SQL chatbot system.

Table 1: Sample Dataset for Model Training.

Master Questions	Master SQL Commands
What colors of shoes are available?	SELECT DISTINCT color FROM wms_products;
How many pairs of red shoes are available?	SELECT SUM(qty) FROM wms_products WHERE color = 'red';
What colors are available for size 42 shoes?	SELECT DISTINCT color FROM wms_products WHERE size = 42;
What sizes are available for red shoes?	SELECT DISTINCT size FROM wms_products WHERE color = 'red'

Additionally, data from the automated warehouse system is used to test query operations and retrieve information to generate responses to user queries.

3.2 Data Preprocessing

The data preprocessing phase is a critical step that directly impacts the chatbot system's performance. This process is divided into four main steps as follows:

Step 1: Intent Assignment for SQL Queries to classify the question types or objectives of the commands. For example, queries that count products (using the COUNT function) are categorized as `ask_stock`. In contrast, queries requesting complete information are classified as `ask_all`, and queries involving summation or quantity aggregation are categorized as `ask_quantity`. Additionally, the following categories include: `ask_date` (date-related queries), `ask_color` (color-related queries), `ask_size` (size-related queries), `ask_model` (model-related queries), and other queries that do not fit into any specific category. This classification enables the model to learn patterns for each query type accurately.

Step 2: Keyword Extraction and Structured Data Extraction from SQL Queries, known as Named Entity Recognition (NER) and Slot Extraction, to identify and store specific information necessary for SQL Template generation. The extracted data is organized into slots such as:

- Color Slot: Product colors (e.g., red, black, brown)
- Size Slot: Product sizes (e.g., 20–45)
- Model Slot: Product models
- Quantity Slot: Desired quantities
- Count Slot: Counting indicators such as "how many items" or "total pairs"

The data processed through Intent Classification and Slot Extraction is compiled into a well-structured dataset that can be used to generate SQL command templates and enhance model performance in subsequent learning and query classification phases.

Step 3: SQL Template Generation to create pro-

totype structures for generating new SQL commands based on user queries. This process enables the model to efficiently learn SQL command structures and patterns that align with query intent. The system employs Regular Expression (RegEx) techniques to search and analyze keywords and SQL command structures in the database. Subsequently, specific data values (e.g., colors, sizes, or product models) are replaced with predefined placeholders, such as:

- Terms like "red", "white", "black" are replaced with the variable `{color}`
- Terms like "42", "38", "40" are replaced with the variable `{size}`

Step 4: Vocabulary Dictionary Definition, which involves compiling specific vocabulary found in the product database to serve as reference data for validating and filtering keywords extracted during the template generation and slot extraction processes. For example, the "shoe colors" vocabulary dictionary contains the following terms: ["pink", "brown", "black", "cream", "khaki"].

When this vocabulary dictionary is used in conjunction with the slot extraction process, the system can identify which terms are related to product attributes. For example, when the term "black" is found in the vocabulary dictionary, it indicates that the user is directly inquiring about product "color" information. The data obtained from these steps is used to train the model in subsequent Intent Classification and Template Selection phases, enabling it to accurately and efficiently select and generate SQL commands in response to user requirements.

3.3 Training Configuration

Training configuration is a critical step that directly impacts the performance and quality of the chatbot system. Proper hyperparameter settings enable the model to efficiently learn data patterns, reduce errors, and enhance prediction accuracy.

3.3.1 Data Splitting

For the task-specific fine-tuning of the chatbot system, the researchers divided the complete dataset of 2,255 entries, consisting of questions and intents, into two subsets:

- Training Set: 1,804 entries (80%)
- Test Set: 451 entries (20%)

The Stratified Split technique was employed to maintain balanced proportions of each intent class across both datasets. Additionally, a random seed was defined to ensure reproducibility of experimental results. Distribution analysis was conducted for both intents and slots to ensure that both datasets exhibit diversity and comprehensively cover all simulated use cases.

3.3.2 Hyperparameter Definition

Appropriate selection and configuration of hyperparameters are critical factors affecting deep model performance, particularly in NLP tasks that require deep learning from linguistic context. This research systematically defines and optimizes hyperparameters across five main categories:

A. Training Strategy Hyperparameter

A two-stage training process was designed to enhance model learning efficiency:

- Stage 1: Training focuses exclusively on the classification head by freezing the backbone (WangchanBERTa) and enabling learning only in the linear layer to prevent catastrophic forgetting and create appropriate feature representations for classification tasks.
- Stage 2: End-to-end fine-tuning occurs during epochs 2–3 by unfreezing both the backbone and head with separate learning rates for each component, allowing the model to perform detailed adjustments suitable for task-specific contexts.

B. Learning Rate Hyperparameter

Learning rates were configured separately according to model architecture to achieve balanced learning:

- Backbone Learning Rate: Set between 1×10^{-5} and 5×10^{-5} for fine-grained adjustments. Since WangchanBERTa is a pre-trained model, lower learning rates reduce the risk of catastrophic forgetting.
- Head Learning Rate: Approximately 5 times higher than the backbone learning rate. Since the classification head requires complete retraining, this configuration enables rapid and efficient adaptation to the new task.

C. Optimization Hyperparameter

This research employs the AdamW optimizer, an enhancement of Adam that separates weight decay from gradient updates, reducing overfitting and increasing training stability. This optimizer is well-suited for transformer-based models such as WangchanBERTa.

D. Regularization Hyperparameter

To enhance generalization capability and reduce overfitting, this research applies three main regularization techniques:

- Gradient Clipping: Limits gradient vector magnitude to Max Norm $= 1.0$ to prevent exploding gradient problems, providing greater training stability, especially in complex deep learning architectures.
- Dropout Regularization: Randomly deactivates neurons during training with a dropout rate of 0.3 (30%) between BERT output and the classification layer to prevent overfitting and enhance learning capability on new data.
- Weight Decay (L2 Regularization): Set to 0.01 to control model weight magnitudes, making the model more robust and reducing excessive fitting to specific data.

E. Training Configuration

Basic training configurations are essential components directly affecting performance and output quality, with the following key settings:

- Number of Epochs: Set to 3 epochs. Since WangchanBERTa is pre-trained, extended training is unnecessary. Limiting epochs reduces the risk of overfitting while enabling early stopping when validation set performance ceases to improve.
- Batch Size: Training batch size set to 16 to accommodate GPU memory constraints and provide stable gradient estimation. The validation batch size is set to 32 to accelerate evaluation, since gradient storage is unnecessary.
- Input Length: MAX_LENGTH set to 64 tokens based on average question length analysis in the dataset, covering most cases while maintaining processing efficiency. The padding strategy uses `padding="max_length"` and `truncation=True` to ensure uniform input length across all data, enabling consistent batch processing.

3.4 Model Architecture

3.4.1 WangchanBERTa

Thai language faces significant limitations in natural language processing resources, resulting in general BERT-based models trained on small datasets having limited performance. Meanwhile, multilingual models such as mBERT or XLM-RoBERTa, trained on large-scale multilingual data, cannot effectively capture Thai-specific characteristics because the proportion of Thai data in their training sets is small. To address these challenges, WangchanBERTa was developed as the largest Thai language model, utilizing the RoBERTa architecture, which is an optimized version of BERT with improved pre-training strategies and hyperparameters. The model was trained on a corpus of over 78.5 GB from diverse sources, including social media platforms, online news articles, web crawling data, Wikipedia, and public knowledge repositories. WangchanBERTa employs a SentencePiece tokenizer designed explicitly for Thai to accommodate its unsegmented writing system, along with enhanced preprocessing procedures, such as sentence segmentation, that account for the Thai language's lack of word boundaries. The model was pre-trained using masked language modeling (MLM) objective to obtain representations suitable for Thai linguistic and semantic characteristics, resulting in WangchanBERTa achieving state-of-the-art results across various Thai NLP tasks such as sentiment analysis, named entity recognition, and question answering [9, 22, 23].

3.4.2 Linear Classifier (LC)

Binary classification problems often face challenges in balancing False Positive (FP) and False Negative (FN) rates. Linear Classifier (LC) has attracted attention for its interpretable structure and customizable objectives. The model operates by creating a linear decision boundary to separate data into classes and can flexibly adjust hyperparameters to prioritize reducing either FP or FN. Logistic Regression (LR) can be viewed as a special case of LC, making this approach more comprehensive [24]. In natural language processing, Linear Classifiers such as Linear Support Vector Machine (Linear SVM) or LR with Term Frequency-Inverse Document Frequency (TF-IDF) features remain highly effective baselines due to their fast training speed, low computational resource requirements, and ability to serve as a sanity check to verify the reasonableness of results from complex deep learning models like BERT, which require intricate fine-tuning [25].

3.5 Performance Evaluation

In evaluating classification model performance, the Confusion Matrix is widely adopted as the primary tool because it displays the counts of correct and incorrect predictions relative to the actual values. The matrix comprises four key components: True Positive (TP), representing the number of positive instances correctly predicted as positive; True Negative (TN), representing the number of negative cases correctly predicted as negative; False Positive (FP), representing the number of negative cases incorrectly predicted as positive; and False Negative (FN), representing the number of positive cases incorrectly predicted as negative [23, 26, 27, 28, 29, 30]. These four values serve as the foundation for calculating various important performance metrics as follows:

3.5.1 Accuracy

Accuracy measures the proportion of correct predictions over the total number of instances, calculated as [23, 26, 27, 28, 29, 30]:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

Accuracy is a fundamental metric for overall model correctness but may be unsuitable for datasets with class imbalance.

3.5.2 Recall

Recall (also known as Sensitivity) measures the model's ability to detect all positive instances, i.e., the number of true positives it identifies from all actual positive instances. It is calculated as [23, 26, 27, 28, 29, 30]:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2}$$

Low Recall indicates a high number of False Negatives, meaning the model frequently fails to detect actual positive instances.

3.5.3 Precision

Precision measures the accuracy of positive predictions, i.e., the fraction of predicted positive instances that are correct. It is calculated as [23, 26, 27, 28, 29, 30]:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3}$$

***Table 2**: Performance Metrics by Intent Category.*

Intents	Precision (%)	Recall (%)	F1-Score (%)
ask_all	92	91	92
ask_color	91	97	94
ask_date	100	100	100
ask_model	100	25	40
ask_quantity	92	98	95
ask_size	100	84	91
ask_stock	99	99	99
other	100	86	92

Low Precision indicates a high number of False Positives, meaning the model frequently makes incorrect positive predictions.

3.5.4 F1-Score

F1-Score represents the harmonic mean between Precision and Recall, used when balanced evaluation of both accuracy and coverage is required. It is calculated as [23, 26, 27, 28, 29, 30]:

$$\text{F-Measure} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

F1-Score achieves high values when both Precision and Recall are simultaneously high, indicating that the model effectively balances the reduction of both False Positives and False Negatives.

3.6 Environment Setup

Hardware Configuration: All experiments were conducted on an MSI laptop with the following specifications: Intel Core i5-11400H processor @ 2.70GHz (6 cores, 12 threads), 16 GB RAM, and 64-bit operating system.

Software Environment: The software stack comprised Python 3.10 for implementation, Visual Studio Code as the development environment, MySQL for database management, and Microsoft Excel for preliminary data analysis. Key libraries included Transformers for WangchanBERTa implementation, Scikit-learn for Linear Classifier, and PyMySQL for database connectivity.

4. RESULTS AND DISCUSSION

4.1 Model Performance

According to the test results, Table 2 presents the performance evaluation of the Intent Classification model. The developed model achieved a Macro F1-Score of 87.84% and an Accuracy of 94.01%, demonstrating that the model can effectively classify questions overall. The model accurately classifies intents including `ask_all`, `ask_color`, `ask_date`, `ask_quantity`, `ask_size`, `ask_stock`, and `other`, with Recall values exceeding 80%, indicating that the model effectively understands the main question

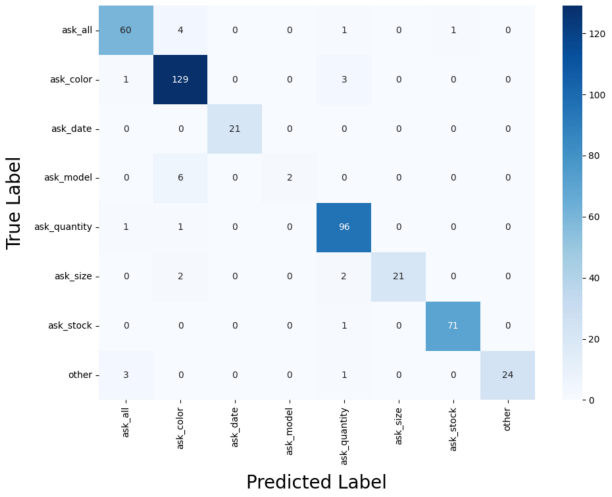


***Fig. 2**: Confusion matrix showing the number of predictions made for each intent class.*

patterns that frequently appear in the system. The `ask_date` intent contains relatively few and less diverse training samples, which may cause the model to rely on memorizing patterns rather than generalizing effectively, potentially indicating overfitting. Due to the limited amount of data available for this intent, further data collection and future model improvements may be required to ensure more robust and generalizable performance.

In contrast, the `ask_model` intent shows limitations. Although the model's Precision reaches 100%, reflecting that the model correctly identifies `ask_model` questions when predicted, the Recall is only 25%, meaning that the model can capture only 1 in 4 of the actual `ask_model` questions. This indicates that many questions cannot be detected as `ask_model`, resulting in an F1-score of only 40%. This problem likely stems from (1) limited training samples, (2) similarity with other intents, or (3) complexity of context requiring specific product code references. The evaluation results show that the model performs well overall, but improvements are still needed for the `ask_model` intent by increasing diverse training data and fine-tuning the model to enable the Chatbot system to respond to questions more comprehensively and accurately.

4.2 Confusion Matrix

Fig. 2 shows the confusion matrix of the test dataset used to evaluate the accuracy of the Intent Classification model. The results demonstrate that the model achieves high prediction accuracy across multiple intents as follows: `ask_color` correctly predicted 129 samples (96.99%), `ask_quantity` correctly predicted 96 samples (97.95%), `ask_stock` correctly predicted 71 samples (98.61%), `ask_all` correctly predicted 60 samples (90.91%), `ask_date` correctly predicted 21 samples (100%), `ask_size` correctly predicted 21 samples (84%), and `other` correctly predicted 24 samples

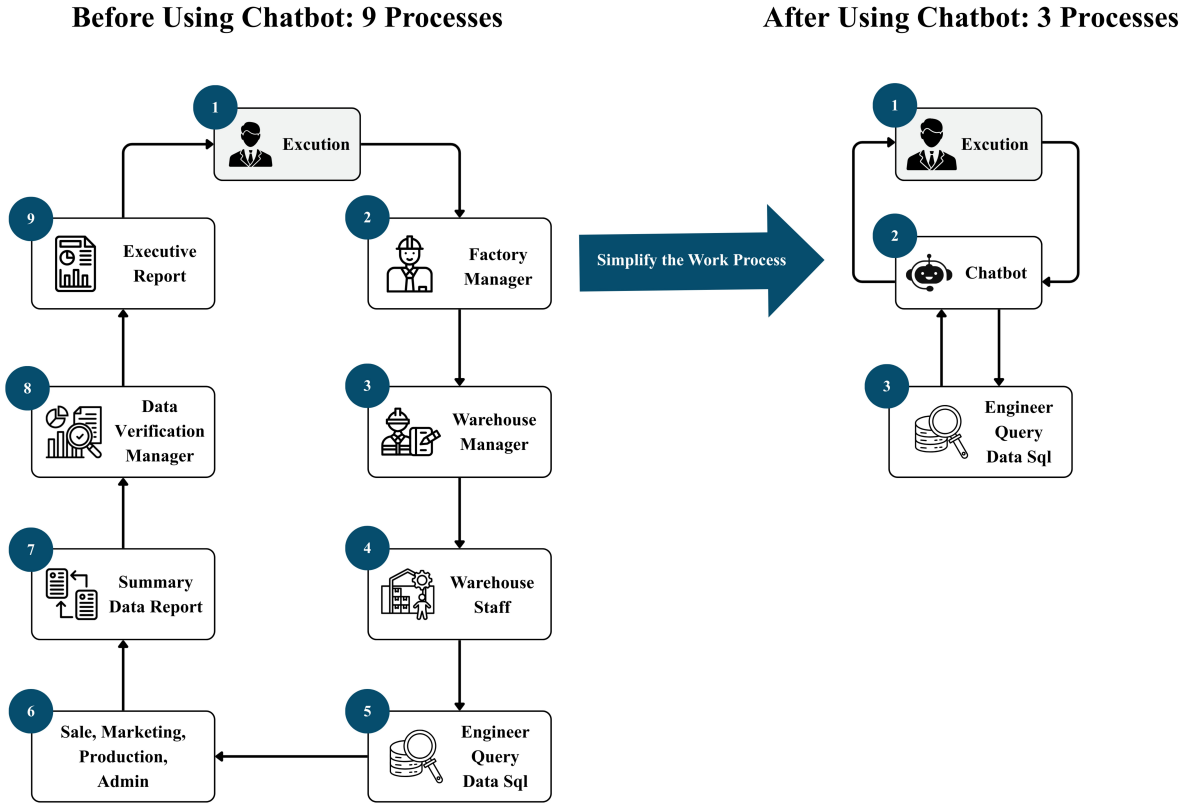


***Fig. 3**: Comparison of Information Retrieval Workflow between Traditional Method and Chatbot-enabled System*

(85.71%). These results indicate that the model can accurately identify questions related to "product color," "stock availability," and "product quantity." However, some misclassifications were observed: `ask_all` was misclassified as `ask_color` 4 times, and `ask_model` was misclassified as `ask_color` 6 times, suggesting the model may struggle when messages contain similar words or lack clear contextual information. Despite some misclassifications in certain intents, the model overall delivers high accuracy (Accuracy = 94.01%) and can be deployed in a real warehouse Chatbot system, with potential improvements through additional training data and fine-tuning for lower-performing intents.

4.3 Chatbot Performance Testing

The researchers randomly selected a set of questions from actual users to cover multiple intent categories, including `ask_all`, `ask_color`, `ask_date`, `ask_model`, `ask_quantity`, `ask_size`, `ask_stock`, and others. The test dataset consisted of 10 questions prepared for each intent category. Each question was then input into the Chatbot system to allow the model to generate appropriate SQL queries for data retrieval. The SQL queries generated by the Chatbot (SQL Result from Chatbot) were compared with the correct SQL queries (True SQL). Finally, warehouse experts or data engineers verified the accuracy of the results by examining the query conditions and the actual data values retrieved from the database. The testing results show that the Chatbot achieves high accuracy in the `ask_color`, `ask_quantity`, and `ask_stock` intents. Conversely, the `ask_model`, `ask_size`, and `other` intents still yield suboptimal results. This is attributed to insufficient training data for these categories and to the need to improve understanding of contextual queries related to temporal data references and specific product code identification in the future.

4.4 System Benefits and Practical Implementation

The implementation of the Chatbot system in the organization provides five key benefits as follows:

4.4.1 Process Simplification

Traditionally, executives had to request information from multiple departments—such as warehouse, production, and sales—and wait for verification from engineers before receiving reports—a process that took approximately 2–4 hours. With the Chatbot, executives can type questions in Thai, such as "How many pairs of Comfort X size 42 remain in stock?" and receive automated responses from the SQL database within 5 seconds, significantly reducing multi-level communication steps, as illustrated in Fig. 3.

4.4.2 Planning Support

Data retrieved by the Chatbot from the WMS database is real-time, enabling executives to immediately visualize

production, sales, and inventory status for rapid strategic decision-making. For example, when executives query "What are the sales figures for Comfort X shoes this week?" and discover rapidly increasing sales due to promotional campaigns, they can immediately decide to increase production capacity or allocate resources without waiting for consolidated reports from multiple departments.

4.4.3 Workload Reduction

Engineers and data verification staff no longer need to perform repetitive SQL queries for executives or general users. Surveys indicate that engineers spend an average of 30–45 minutes per inquiry and receive approximately 5–10 requests daily. With Chatbot handling these tasks, engineers can focus on in-depth analysis, system development, or tasks that require greater expertise.

4.4.4 Cost and Time Efficiency

The system reduces inter-departmental communication time, report waiting periods, and errors from multi-layer communication. Analysis shows that information request time has decreased from an average of 2–4 hours to just 5 seconds, resulting in over 99% time savings. Additionally, it reduces personnel costs associated with data intermediation roles.

4.4.5 Data Accuracy Enhancement

The Chatbot retrieves data directly from the WMS SQL database, minimizing risks of data discrepancies from manual re-entry or multi-person transfers. Direct retrieval from a single source of truth ensures users receive accurate, consistent, and up-to-date information. A comparison of workflow processes between traditional methods and Chatbot-enabled approaches is shown in Fig. 3.

5. CONCLUSIONS

This research presents the design and development of a Thai language Chatbot for querying data from a WMS in the footwear manufacturing industry. The system integrates NL2SQL techniques with natural language processing using WangchanBERTa and a Linear Classifier for Intent Classification, Slot Extraction, and SQL Template generation for real database queries. The system was developed and tested on a dataset of 2,255 samples, split into training and testing sets at an 80:20 ratio. Evaluation covered both classification metrics and the Chatbot's actual performance in SQL generation and database querying.

Intent Classification Results: The developed model achieved a Macro F1 Score of 87.84% and an Accuracy of 94.01%, demonstrating strong overall intent classification performance. The model performed exceptionally well on specific intents: `ask_color` correctly predicted 129 samples (96.99%), `ask_quantity` correctly predicted 96 samples (97.95%), `ask_stock` correctly predicted 71 samples (98.61%), `ask_all` correctly predicted 60 samples (90.91%), `ask_date` correctly predicted 21 samples (100%), `ask_size` correctly predicted 21 samples (84%), and `other` correctly predicted 24 samples (85.71%). All these intents achieved Recall values above 80%. However, a critical weakness was identified in the `ask_model` intent, with a Recall of 25%, potentially due to limited training samples, similarity to other intents, or contextual complexity that requires specific product code references, necessitating additional training data or model improvements to better handle questions.

NL2SQL Testing on Real Data: Evaluation of SQL query accuracy for each intent revealed that `ask_color`, `ask_quantity`, and `ask_stock` achieved perfect results (100% SQL Accuracy), while `ask_date` = 0%, `ask_model` = 10%, `ask_size` = 50%, and `other` = 0%. This highlights challenges in handling questions that require temporal references, specific product codes, and complex contexts. The primary issues stem from insufficient training data and limited diversity in question patterns within these classes.

Research Limitations: Performance degradation in certain intents results from (1) data imbalance, particularly for intents with few samples, (2) limited diversity in question patterns, especially for temporal and product code queries, and (3) contextual complexity requiring multi-dimensional data references, which reduces SQL conversion and result retrieval accuracy. To enhance system performance and coverage, data quality and quantity should be improved by adding samples for problematic intents (`ask_model`, `ask_date`, `other`), emphasizing contextual diversity and real-world use cases, including data augmentation techniques. Additionally, Intent structure and Slot Extraction should be refined by considering more granular intent definitions to reduce ambiguity and by developing domain-specific NER models with product code dictionaries and date format patterns to improve slot-filling accuracy. Implementing quality control mechanisms using confidence thresholds for expert validation in low-confidence cases and incorporating feedback for continuous model improvement (Human-in-the-loop learning) will enhance system reliability. Finally, real-world testing and tuning should be conducted to evaluate performance in actual use cases and to collect additional user data for sustainable development.

Overall, this study demonstrates that the Wangchan BERTa-based approach, combined with template/slot-based NL2SQL processes, can effectively build a Thai-language Chatbot for WMS data querying, particularly for questions that reference clear attributes such as color, size, quantity, and stock. This reduces workflow steps, employee workload, and time costs, while increasing data accuracy and supporting real-time data-driven planning. However, for enterprise-level deployment readiness, it is essential to increase the volume and diversity of training data, optimize the intent structure, and enhance quality control mechanisms to ensure sufficient accuracy and robustness for practical future applications.

ACKNOWLEDGEMENT

The authors gratefully acknowledge the Center of Multidisciplinary Innovation Network Talent (MINT Center), Faculty of Interdisciplinary Studies, Department of Technology and Engineering, Khon Kaen University, Nong Khai Campus, Thailand, for providing research knowledge, facilities, and computational resources. The authors would like to acknowledge Smartech Industry Co., Ltd. for providing industrial data, research concepts, and domain expertise that supported this study. The authors also acknowledge their advisors and co-advisor for their valuable guidance and constructive suggestions throughout this research. Finally, the authors acknowledge their institutions for fostering a research-friendly environment that enabled this research.

AUTHOR CONTRIBUTIONS

The manuscript was written with contributions from all authors. They discussed the results and commented on the manuscript. All authors approved the final version of the manuscript.

REFERENCES

[1] S. Ali and Y. Xie, "The impact of Industry 4.0 on organizational performance: The case of Pakistan's retail industry," *European Journal of Management Studies*, vol. 26, no. 2/3, pp. 63–86, 2021.

[2] A. N. Anang, P. O. Obidi, A. O. Mesogboriwon, J. O. Obidi, M. Kuubata, and D. Ogunbiyi, "The role of artificial intelligence in Industry 5.0: Enhancing human-machine collaboration," *World Journal of Advanced Research and Reviews*, vol. 24, no. 2, pp. 380–400, Nov. 2024.

[3] M. Ghobakhloo, "Industry 4.0, digitization, and opportunities for sustainability," *Journal of Cleaner Production*, vol. 252, Art. no. 119869, Apr. 2020.

[4] A. Jarašūnienė, K. Čižiūnienė, and A. Čereška, "Research on impact of IoT on warehouse management," *Sensors*, vol. 23, no. 4, p. 2213, 2023.

[5] F. Krause et al., "Managing human-AI collaborations within Industry 5.0 scenarios via knowledge graphs: Key challenges and lessons learned," *Frontiers in Artificial Intelligence*, vol. 7, Art. no. 1247712, Nov. 2024.

[6] P. M. Mah, I. Skalna, and J. Muzam, "Natural language processing and artificial intelligence for enterprise management in the era of Industry 4.0," *Applied Sciences*, vol. 12, no. 18, p. 9207, 2022.

[7] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, and J. Gao, "Deep learning–based text classification: A comprehensive review," *ACM Computing Surveys*, vol. 54, no. 3, Art. no. 62, pp. 1–40, Apr. 2021.

[8] G. Caldarini, S. Jaf, and K. McGarry, "A literature survey of recent advances in chatbots," *Information*, vol. 13, no. 1, p. 41, 2022.

[9] L. Lowphansirikul, C. Polpanumas, N. Jantrakulchai, and S. Nutanong, "Wangchanberta: Pretraining transformer-based Thai language models," *arXiv preprint arXiv:2101.09635*, 2021.

[10] M. A. Ardanta, A. Fauzi, P. Patimah, F. Khadijah, S. R. Yunandi, and A. M. Ghowe, "Human-AI collaboration in supply chain Industry 5.0 to build a human-centered autonomous ecosystem," *Siber International Journal of Digital Business (SIJDB)*, vol. 2, no. 4, pp. 357–368, 2025.

[11] M. Pournader, H. Ghaderi, A. Hassanzadegan, and B. Fahimnia, "Artificial intelligence applications in supply chain management," *International Journal of Production Economics*, vol. 241, p. 108250, 2021.

[12] E. Vann Yaroson, A. Abadie, and M. Roux, "Human-artificial intelligence collaboration in supply chain outcomes: The mediating role of responsible artificial intelligence," *Annals of Operations Research*, vol. 354, no. 1, pp. 35–69, Nov. 2025.

[13] Y. Zhang, R. Y. K. Lau, J. D. Xu, Y. Rao, and Y. Li, "Business chatbots with deep learning technologies: State-of-the-art, taxonomies, and future research directions," *Artificial Intelligence Review*, vol. 57, no. 5, Art. no. 113, May 2024.

[14] M. Zulqarnain, R. Ghazali, Y. M. M. Hassim, and M. Rehan, "A comparative review on deep learning models for text classification," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 19, no. 1, pp. 325–335, 2020.

[15] K. Majhadi and M. Machkour, "Chat-SQL: Natural language text to SQL queries based on deep learning techniques," *Journal of Theoretical and Applied Information Technology*, vol. 102, no. 12, pp. 5052–5061, 2024.

[16] X. Liu et al., "A survey of Text-to-SQL in the era of LLMs: Where are we, and where are we going?," *IEEE Transactions on Knowledge and Data Engineering*, vol. 37, no. 10, pp. 5735–5754, Oct. 2025.

[17] H. Dihingia, S. Ahmed, D. Borah, S. Gupta, K. Phukan, and M. K. Muchahari, "Chatbot implementation in customer service industry through deep neural networks," in *2021 International Conference on Computational Performance Evaluation (ComPE)*, 2021, pp. 193–198.

[18] T. Scholak, N. Schucher, and D. Bahdanau, "PICARD: Parsing incrementally for constrained auto-regressive decoding from language models," *arXiv preprint arXiv:2109.05093*, 2021.

[19] R. G. Richey Jr., S. Chowdhury, B. Davis-Sramek, M. Giannakis, and Y. K. Dwivedi, "Artificial intelligence in logistics and supply chain management: A primer and roadmap for research," *Journal of Business Logistics*, vol. 44, no. 4, pp. 532–549, 2023.

[20] M. Verma, "Integration of AI-based chatbot (ChatGPT) and supply chain management solution to enhance tracking and queries response," *International Journal for Science and Advance Research in*

Technology, vol. 9, no. 2, pp. 60–63, Feb. 2023.

[21] P. Wongpraomas, C. Soomlek, W. Sirisangtragul, and P. Seresangtakul, "Thai question-answering system using pattern-matching approach," in *Proc. 2022 1st Int. Conf. Technology Innovation and Its Applications (ICTIIA)*, 2022, pp. 1–5.

[22] P. Harnmetta and T. Samanchuen, "Sentiment analysis of Thai stock reviews using transformer models," in *Proc. 2022 19th Int. Joint Conf. Computer Science and Software Engineering (JCSSE)*, 2022, pp. 1–6.

[23] M. Nokkaew et al., "Analyzing online public opinion on Thailand-China high-speed train and Laos-China railway mega-projects using advanced machine learning for sentiment analysis," *Social Network Analysis and Mining*, vol. 14, no. 1, Art. no. 15, Dec. 2024.

[24] H. Jeong and B. Zou, "Linear classifier models for binary classification," *Variance*, vol. 18, 2025.

[25] Y. C. Lin, S. A. Chen, J. J. Liu, and C. J. Lin, "Linear classifier: An often-forgotten baseline for text classification," *arXiv preprint arXiv:2306.07111*, 2023.

[26] S. Agrawal, S. K. Jain, S. Sharma, and A. Khatri, "COVID-19 public opinion: A Twitter healthcare data processing using machine learning methodologies," *International Journal of Environmental Research and Public Health*, vol. 20, no. 1, Art. no. 432, Jan. 2023.

[27] M. Nokkaew et al., "Hidden emotional trends on social media regarding the Thailand–China high-speed railway project: A deep learning approach with ChatGPT integration," *Social Network Analysis and Mining*, vol. 14, no. 1, Art. no. 175, Dec. 2024.

[28] A. Tharwat, "Classification assessment methods," *Applied Computing and Informatics*, vol. 17, no. 1, pp. 168–192, Jan. 2021.

[29] A. K. Sharma and R. Kumar, "IoT malware detection and mitigation in AMQP simulated environment," *ECTI Transactions on Computer and Information Technology (ECTI-CIT)*, vol. 18, no. 4, pp. 469–480, 2024.

[30] R. Tanawongsuwan, S. Phongsuphap, and P. Mongkolwat, "Evaluating trust in CNN transfer learning with flower image classification via heatmap-based XAI," *ECTI Transactions on Computer and Information Technology (ECTI-CIT)*, vol. 19, no. 3, pp. 392–405, 2025.



Thaninrath Witoontharaphong is the Founder and Chief Executive Officer of TW Engineering & Programming Co., Ltd., Thailand. His expertise includes the design and development of Warehouse Management Systems (WMS), Enterprise Resource Planning (ERP) systems, and AI-enabled industrial software solutions. He is currently pursuing the Master of Engineering degree in Electrical and Computer Engineering at Khon Kaen University, Thailand. His research interests include artificial intelligence for industrial applications, natural language processing, machine learning, and intelligent systems for smart manufacturing.



Manussawee Nokkaew received the Ph.D. degree in Electrical and Computer Engineering from Khon Kaen University, Thailand, in 2024. From 2015 to 2025, he worked as an analyst and consultant at Software Park Thailand under the National Science and Technology Development Agency (NSTDA). During this period, his work focused on managing projects to empower software and digital entrepreneurs in accessing domestic and international markets, and driving digital transformation among industrial enterprises. Currently, he is a Lecturer in the Computer and Information Science Program, Technology and Engineering Sector, at Khon Kaen University, Nong Khai Campus. His research interests include artificial intelligence (AI), particularly generative AI and natural language processing (NLP), as well as machine learning, deep learning, big data analytics, and sustainability.



Chayada Surawanitkun received the B.Eng. degree in Electrical Engineering (1st class honor) and the Ph.D. degree in Electrical Engineering from Khon Kaen University, Thailand, in 2009 and 2013, respectively. She was granted the B.Eng. Scholarship from the Industrial/University Cooperative Research Center in HDD Component, supported by the Ministry of Science and Technology, and a Ph.D. scholarship from the Royal Golden Jubilee Ph.D. Program under The Thailand Research Fund. She is Program Chair of Master's and Doctoral programs in Electrical and Computer Engineering of the Faculty of Interdisciplinary Studies, Department of Technology and Engineering, Khon Kaen University, Nong Khai Campus, Thailand. Her research interests include power systems, Smart IoTs, data storage technology, magnetic random-access memory, thermal analysis of nanodevices, nanotechnology, renewable and sustainable energy, and energy management systems.



Kanda Sorn-In is an Assistant Professor in the Computer and Information Science Program, Department of Technology and Engineering, Faculty of Interdisciplinary Studies, Khon Kaen University, Thailand. She received her B.Sc. and M.Sc. degrees in Computer Science and her Ph.D. in Information Studies from Khon Kaen University. Her research interests include data analysis, information systems design, artificial intelligence, neural networks, business intelligence, recommender systems, and sentiment analysis.



Nongram Mueanrit received her Ph.D. in Information Studies from Khon Kaen University, Thailand, in 2013. She is currently serving as an Assistant Professor in the Department of Technology and Engineering, Faculty of Interdisciplinary Studies, Khon Kaen University. Her research interests encompass a broad spectrum of interdisciplinary domains, including information behavior analytics, consumer behavior analysis, user experience design, business data analytics, and smart tourism systems. She has contributed to the academic community through publications in internationally recognized peer-reviewed journals, including Energy, among others.



Satit Kravenkit received the B.Sc., M.Sc., and Ph.D. degrees in Computer Science from Khon Kaen University, Thailand, in 2001, 2006, and 2017, respectively. From 2004 to 2006, during his graduate studies, he worked as a researcher in semantic web technology with the Department of Computer Science, Khon Kaen University. He later served as a lecturer with the Department of Computer and Information Science, Faculty of Interdisciplinary Studies, Khon Kaen University, Nong Khai Campus. From 2023 to 2026, he worked with the College of Computing at Khon Kaen University. His Ph.D. thesis focused on developing a traceability system compliant with the GS1 EPCglobal Network for cloud-based Electronic Product Code Information Services (EPCIS). His research interests include radio frequency identification (RFID), cloud computing, product traceability and recall in supply chains, blockchain technology, and cybersecurity.



Thalerngsak Wiangwiset received the B.Eng. degree in electrical and computer engineering from Naresuan University, Thailand, in 2004, the M.Sc. degree in computer science from Khon Kaen University, Thailand, in 2008, and the Ph.D. degree in electrical and computer engineering from Khon Kaen University, Thailand, in 2025. He is currently a System Analyst and Developer with the National Telecom Public Company Limited (NT), Thailand. His research interests include cybersecurity, data science, agentic AI, and software engineering. He has received multiple innovation and outstanding research awards from NT and Khon Kaen University.



Apirat Siritaratiwat received the B.Eng. degree in Electrical Engineering from Khon Kaen University, Thailand, in 1992. After working in industry for a few years, he joined the Department of Electrical Engineering, Khon Kaen University (KKU) in 1994 and continued his postgraduate study at the University of Manchester, U.K., between 1995 and 1999. He has done research in HDD and is one of the pioneer researchers of ESD/EOS and EMI in recording heads with the KKU-Seagate Cooperation Research Laboratory, Faculty of Engineering, Khon Kaen University. Currently, he works at the Department of IoT and Information Engineering, Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang, Bangkok. His research interests are ESD/EOS, EMI, renewable energy and energy management system.