Article:

Yuan, Z., Liu, K., Li, S. et al. (4 more authors) (2025) PEZEGO: A precision agriculture system based on large language models and internet of things for pest management. IEEE Internet of Things Journal. ISSN: 2327-4662

https://doi.org/10.1109/JIOT.2025.3586374

PEZEGO: A Precision Agriculture System Based on Large Language Models and Internet of Things for Pest Management

Zhipeng Yuan, Kang Liu, Shunbao Li, Ruoling Peng, Daniel Leybourne, Nasamu Musa, and Po Yang

***Abstract*—Pests significantly threaten global agricultural production, which causes severe yield losses through feeding and virus transmission. To mitigate yield losses caused by pests, timely and precise pest management practices are critical. Although previous efforts have advanced automated solutions for real-time environmental monitoring in agriculture, implementing precise pest management decision-making and suggestion generation remains challenging due to complex reasoning processes in practice. In response, an enhanced pest management system, PEZEGO, is proposed to provide precise management suggestions through multi-modal environmental data, a fine-tuned open vocabulary detector (OVD), and large language models (LLMs). Specifically, a mobile application and low-cost IoT devices are developed to capture images and environmental information. A hybrid convolutional low-rank adaptation method (HCLoRA) is proposed to fine-tune pre-trained OVDs, enabling zero-shot pest detection for converting images to pest species and quantity information. In addition, a structured data-based retrieval augmented generation workflow for LLMs is proposed to provide precise pest management suggestions through automatically extracted agriculture management knowledge and chain-of-thought. The effectiveness of PEZEGO is validated in a case study of pest management in the UK, including pest detection in field scenarios and management suggestion generation. Compared to advanced model fine-tuning methods, HCLoRA achieves the highest detection performance with 0.1759 AP^h for YOLOWorld on pest detection. Additionally, the proposed structured data-based retrieval augmented generation workflow obtains 68.7% average Entity-level F1 score for knowledge extraction and 77.33% accuracy for pest management suggestion generation. Eventually, a user-friendly mobile application demonstrates the practical effectiveness of the proposed PEZEGO system.**

***Index Terms*—Precision agriculture, Internet of Things, Large language model, Retrieval augmented generation, Pest management.**

This paper was supported by Innovate UK under projects 10050919, 10092051, and 10123768. *(Corresponding author: Po Yang)*

Zhipeng Yuan, Shunbao Li, Ruoling Peng, and Po Yang are with the School of Computer Science, University of Sheffield, Sheffield, S10 2TN, United Kingdom (e-mail: zhipeng.yuan@sheffield.ac.uk; shunbao.li@sheffield.ac.uk; rpeng5@sheffield.ac.uk; po.yang@sheffield.ac.uk).

Kang Liu is with the Department of Aeronautical and Aviation Engineering, Hong Kong Polytechnic University, Hong Kong 999077, China, (e-mail: k1liu@polyu.edu.hk)

Daniel Leybourne is with the Department of Evolution, Ecology and Behaviour, University of Liverpool, Liverpool, L69 7ZB, United Kingdom (e-mail: daniel.leybourne@liverpool.ac.uk).

Nasamu Musa is with the Department of Soils, Crops and Water, RSK ADAS Ltd High Mowthorpe, Malton, YO17 8BP, United Kingdom (e-mail: nasamu.musa@adas.co.uk)

I. INTRODUCTION

With global warming, the increasing incidence of pest outbreaks seriously threatens food security worldwide [1]. Statistically, pests are responsible for over 20% of global annual yield loss [2]. To mitigate the loss caused by pests, agronomists are employed in traditional agriculture to estimate pest population densities and to determine management strategies by manual observations and extensive management experience [3]. However, the timeliness and labour intensity of traditional agriculture hinder the effectiveness and sustainability of management strategies. As a response to the limitations in traditional agriculture, precision agriculture has been proposed to provide accurate and timely automated agriculture management support with data-driven technologies.

Over the past decade, various works in precision agriculture explored the feasibility of implementing automated pest [4] or disease [5] monitoring by combining IoT sensors with advanced deep learning models. In addition, automated spraying systems were developed based on drones or mechanical equipment for efficient chemical control [6]. However, these previous works ignored the complex reasoning process and sustainable strategies in agricultural practice. Especially, the complex influence of environment on sustainable practice, such as climate [7], location [8], and economic benefits, poses a challenge for providing sustainable agricultural support.

In this work, we implement a pest management system, PEZEGO, which addresses the reasoning challenges of providing sustainable management support by introducing large language models (LLMs) in a precision agriculture system. Inspired by the text comprehension and reasoning ability of LLMs, LLMs play the role of reasoners in PEZEGO, using environmental information and agricultural knowledge to provide sustainable pest management suggestions. Specifically, PEZEGO consists of Internet of Things (IoT) sensors, open vocabulary detectors (OVDs) [9], LLMs [10], and a cloud computing platform. IoT sensors are utilised to capture images and environmental information as system input for analysis. The OVD provides a zero-shot detection solution by integrating a text encoder in object detection models for estimating the density of numerous pest species from captured images. Eventually, a structured data-based retrieval augmented generation (SRAG) workflow is introduced to provide accurate pest management suggestions based on environmental information, chain-of-thought (CoT), and retrieved agricultural knowledge

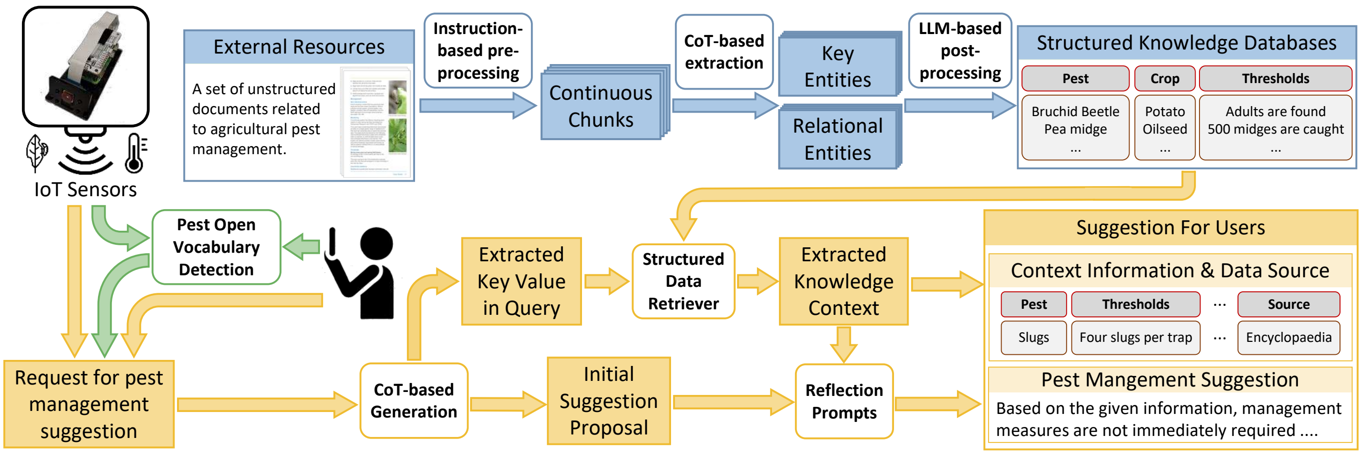


Fig. 1. Workflows of PEZEGO covering zero-shot pest detection, knowledge extraction, and suggestion generation. The green arrows present the data flow for pest detection, which obtains images from sensors or mobile phones to estimate pest species and densities. The blue rectangles show the workflow of the knowledge extraction method, which coverts unstructured external resources to structured knowledge databases. The yellow rectangles demonstrate the suggestion generation in the SRAG workflow. The rounded rectangles in the figure indicate methods. Right-angled rectangles represent data.

data. An overall workflow of PEZEGO is shown in Figure 1.

The novelty of PEZEGO involves three LLM-inspired methods for supporting pest management. Firstly, a hybrid convolutional low-rank adaptation method (HCLoRA) is proposed to fine-tune OVDs for adapting pest detection tasks with few trainable parameters. To the best of our knowledge, this is the first fine-tuned OVD for pest detection. Secondly, a knowledge extraction method based on prompt engineering is implemented to automate the agricultural knowledge base construction, which has the potential to construct an up-to-date agricultural knowledge base in a structured textual format. Last, a suggestion generation method is proposed to address the hallucination of LLMs. The proposed knowledge extraction and suggestion generation methods constitute the SRAG workflow. With the above methods, PEZEGO demonstrates the effectiveness of LLMs in supporting sustainable agriculture. The contributions of this work are summarised as follows.

1) A fine-tuning method, HCLoRA, is proposed for OVDs, which implement real-time zero-shot pest detection without the requirement of constructing a large-scale pest detection dataset.
2) A SRAG workflow, consisting of knowledge extraction and suggestion generation methods, is proposed to generate accurate pest management suggestions based on environmental information and structured knowledge.
3) A case study of UK pest management is completed to qualitatively and quantitatively validate PEZEGO with proposed methods, which outperform state-of-the-art methods in terms of detection and suggestion accuracy.

II. Related work

This section reviews relevant studies on precision agriculture systems and LLMs to provide a comprehensive research context for this work.

A. Precision Agriculture System

Precision agriculture systems are dedicated to increasing yields and reducing environmental pollution in agriculture through IoT sensors and data analysis. Related works started with the monitoring of farms and gradually evolved into an automated system. For preliminary studies, a real-time pest monitoring solution [11] was proposed through a light trap and an optimised deep learning model. This work demonstrated balanced performance in terms of speed and accuracy, achieving 71.3% mean average precision (mAP) for 24 classes. In addition to image-based solutions, a fuzzy logic system was proposed to predict crop pest breeding for rice and millet through weather information captured by IoT-based monitoring infrastructure [7].

Based on the pest and disease detection ability, several studies have implemented automated pesticide spraying through unmanned robotic vehicles or drones for effective management practices. For example, a drone-based automated spaying system [12] was designed for managing Tessaratoma papillosa on longan crops. This system integrates YOLOv3 to locate pests in real-time for planning the optimised pesticide spraying routes and areas. Although the above efforts have demonstrated the effectiveness of precision agriculture for farm monitoring and management, these works simplify complex reasoning in agricultural management, such as the integration of environmental information, economic thresholds, and sustainable management strategies. Therefore, the implementation of sustainable management strategies is still a challenge for current precision agriculture systems.

B. Large Language Models

LLMs refer to text generation models with more than 10 million trainable parameters [10], demonstrating the potential to solve the challenges in precision agriculture with text comprehension and reasoning abilities. Most LLMs employ a transformer model or its variants [13] that are pre-trained on large-scale data via elaborate self-supervision strategies and transferred to specific downstream tasks through supervised fine-tuning. Compared with previous natural language processing (NLP) models [14] training for a single downstream task, LLMs have demonstrated attracted learning capabilities, including instruction learning [15], in-context learning [16], reasoning [17], and cross-modal generalised representation [9].

Instruction learning refers to zero-shot methods of guiding LLMs to complete downstream tasks through task instruction. A common task instruction consists of a task description, supplementary information, the definition of anticipated input format, and anticipated output format [15]. In contrast, in-context learning acts as few-shot learning methods that guide LLMs to complete downstream tasks by identifying hidden patterns from target task samples without modifying the parameters of LLMs [16]. In-context learning has been widely used in numerous NLP tasks, such as information extraction task [18] and machine translation task [19]. In addition, LLMs have demonstrated reasoning ability to break down complex problems into simple subproblems and solve them sequentially, such as CoT [20], which stimulates intermediate reasoning for downstream text tasks. In addition to text tasks, the Platonic Representation Hypothesis in LLMs [21] introduced ability of LLMs for cross-modal transfer learning. Inspired by this hypothesis, OVDs were proposed for zero-shot object detection, by guiding visual model training [9].

The above capabilities of LLMs have attracted researchers to explore the feasibility of LLMs in knowledge-intensive fields. For instance, LLMs demonstrated surprising potential in primary diabetes care [22] and legal judgment [23]. For agriculture, some efforts explored the effectiveness of LLMs in agriculture by constructing datasets [24], providing evaluation methods [25], and providing auxiliary detection [26]. However, providing sustainable pest management suggestions is still underexplored. In addition, the hallucination issue in LLMs threatens the reliability of LLMs in agriculture. To address this problem, a SRAG workflow is proposed based on an automated knowledge extraction method for efficient and accurate pest management suggestion generation.

III. PEZEGO System

PEZEGO is designed with a client-server architecture, as shown in Fig. 2. In this architecture, IoT sensors and a mobile application serve as clients for capturing system input data and accessing system function through user-friendly human-machine interaction. The server is deployed on a cloud computing platform with a microservice framework to ensure system availability and scalability. The microservice framework manages four system services with corresponding data storage, including farm management, pest detection, knowledge extraction, and suggestion generation services. Specifically, the farm management service in PEZEGO provides basic management functions, covering field, crop, farmer, and practice information. The pest detection service implements an image-based pest detection function through a fine-tuned OVD to provide pest information for suggestion generation. Knowledge extraction and suggestion generation services are two LLM-based services for reliable suggestion generation. The knowledge extraction service automatically extracts knowledge from unstructured textual documents to constitute structured data for supporting suggestion generation with reliable knowledge. The

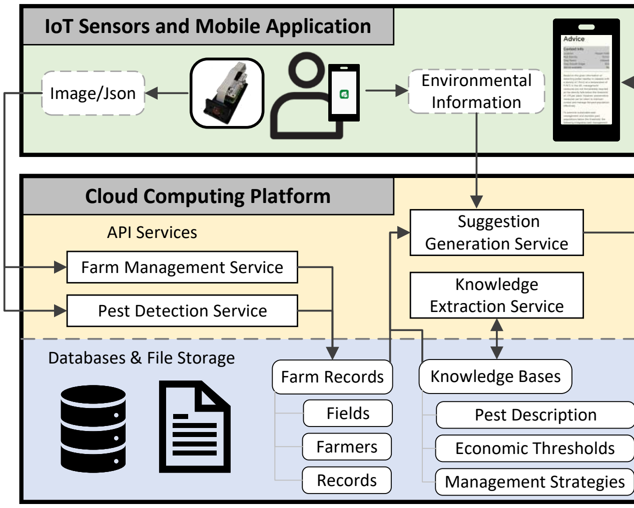


Fig. 2. PEZEGO system architecture. The IoT sensors and a mobile application are clients for collecting images and environmental information to support suggestion generation. The cloud computing platform provides a unified resource management solution for farm management, suggestion generation, and knowledge extraction with four API services and two databases. The black arrowed line demonstrate the data flow of this system. Captured images is sent to detection service for conversion to tabular pest data, which is sent to suggestion generation service with environmental data.

structured data is saved in a database to solve the hallucination issue of LLMs. The suggestion generation service provides reliable suggestions based on retrieval and CoT. In this system, images and environmental data captured by IoT devices or mobile applications are transmitted to the server. Specifically, the image data is processed by the pest detection service to be converted into tabular pest data, which is sent to the suggestion generation service with captured environmental data for obtaining suggestions. In this section, specific hardware and algorithm designs are demonstrated.

A. Hardware Device Design

The IoT device for information collection, as shown in Fig. 3, consists of a microprocessor unit, a camera module, and environmental sensors. The microprocessor unit of this device is implemented by Raspberry Pi Zero 2 W, which is a single-board computer with a quad-core 64-bit ARM CPU and wireless LAN. The whole device is powered via a micro-USB socket on Raspberry Pi Zero 2 W. In this device, a high-resolution autofocus camera module, which has a Sony IMX519 sensor with 4656×3496 pixel resolution and a built-in autofocus motor, is connected to the microprocessor through a camera serial interface. The camera module is positioned to face pest activity zones, such as leaf surfaces or plant stems, for periodically capturing images. In addition, environmental sensors, BME280 and LTR-559, on the Enviro board are connected to Raspberry Pi Zero 2 W through GPIO pins. BME280 can capture environmental data for temperature, barometric pressure, and humidity, which are key contextual information for pest management. LTR-559, as a light sensor, is utilised to detect ambient light to determine if it is an appropriate time for image capture. A data collection program is installed

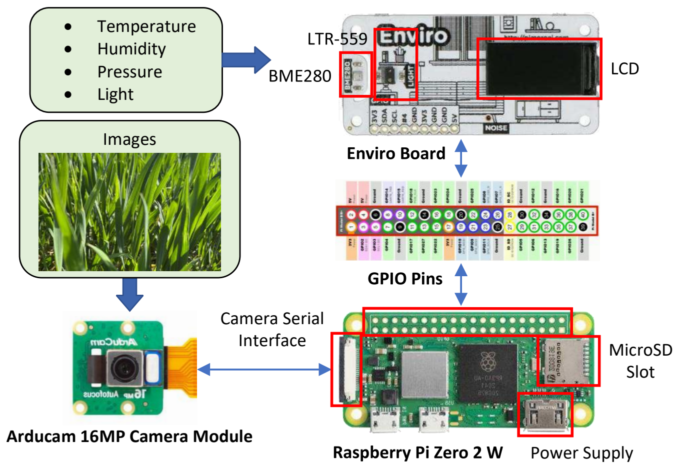


Fig. 3. IoT sensor component connection diagram. This device consists of a Raspberry Pi Zero, a camera module and an Enviro board for image and environmental data collection.

in the Raspberry Pi, which regularly captures images and environmental data and uploads them to the PEZEGO system over the network or stores them locally.

B. Fine-tuning for Open Vocabulary Detectors

OVD [9], which achieves zero-shot detection for arbitrary objects through text comprehension, presents a potential solution for the challenge of constructing large comprehensive datasets covering various pest species. A general framework of OVDs, as shown in Fig. 4, includes a text encoder, image backbone, feature fusion module, classification head, and bounding box head. Compared with object detection models, it integrates a text encoder to provide generic textual features that are combined with visual features through a feature fusion module. The fused features are fed into classification and bounding box headers to output object detection results with text guidance. The classification header is implemented by text-region comparisons [9], which determines a classification result by the similarity between fused features and textual features. OVDs are pre-trained on large-scale text-region datasets for learning generalised text comprehension and object detection abilities. However, the performance of OVDs significantly degrades when the domain of target objects changes, especially for pest detection. To address this issue, we propose HCLoRA, inspired by low-rank adaptation (LoRA) fine-tuning [27], to fine-tune OVDs on pest detection datasets.

For OVD fine-tuning, HCLoRA modules, consisting of two branches, replace some neural network layers in the image backbone and feature fusion module. A base feature branch $F_p()$ inherits the original pre-trained weights $\mathcal{W}_p$ in the OVD, which is frozen during the fine-tuning process to maintain generalised detection ability. Another fine-tuned feature branch $F_t()$ employs an encoder, a set of convolutional layers with different kernel sizes, and a decoder, as trainable parameters during the fine-tuning process, to learn residual features between the general object and pest for adapting a pest detection task. The encoder and decoder work on channel-wise features. Convolutional layers deal with spatial features, which implement separated channel-wise and spatial feature processing with multiple visual fields. The trainable parameters of the encoder, set of convolutional layers, and decoder are represented by $\mathcal{W}_e$, $\{\mathcal{W}_c^n\}_{n=1}^3$, and $\mathcal{W}_d$, respectively. The computational process of HCLoRA modules is defined as

$$\begin{aligned} F_t(f^i) &= \mathcal{W}_d \cdot C(\mathcal{W}_e f^i, \{\mathcal{W}_c^n \mathcal{W}_e f^i\}_{n=1}^3) \\ F_p(f^i) &= \mathcal{W}_p f^i \\ f^o &= F_p(f^i) + F_t(f^i), \end{aligned} \tag{1}$$

where f^i and f^o are the input and output features of the HCLoRA module, respectively. $C()$ is a concatenation function in channel dimension for a fusion of features from convolution layers with multiple receptive fields. $\mathcal{W}_e f^i$ is the feature processed by the encoder. $\{\mathcal{W}_c^n \mathcal{W}_e f^i\}_{n=1}^3$ is three features from three convolutional layers, as shown in Fig. 4.

During the fine-tuning process, trainable parameters in HCLoRA modules are optimised on a pest detection dataset through a stochastic gradient descent algorithm with a detection loss function defined as,

$$L = \lambda_{cls} \cdot L_{cls} + \lambda_{iou} \cdot L_{iou} + \lambda_{dfl} \cdot L_{dfl}, \tag{2}$$

where λ_{cls}, λ_{iou}, and λ_{dfl} are weights for classification loss L_{cls}, object intersection over union loss L_{iou}, and distributed focal loss L_{dfl}. Specifically, the classification loss L_{cls} is implemented by a binary cross-entropy loss function to measure classification accuracy. The intersection over union loss L_{iou} and distributed focal loss L_{dfl} are utilised to measure the precision of bounding boxes. Other pre-trained parameters, such as $\mathcal{W}_p$ in HCLoRA modules and unreplaced original network layers, are frozen during the fine-tuning process.

C. Knowledge Extraction Method

In response to the lack of structured agriculture knowledge bases, we propose a three-stage knowledge extraction method to support suggestion generation and mitigate the hallucination issue of LLMs. The proposed extraction method structures knowledge by extracting named entities and textual descriptions with relationships from unstructured documents to provide reliable external information for suggestion generation. To simplify the description, entities are used to include named entities and textual descriptions in subsequent articles.

Formally, the input of the knowledge extraction includes a set of unstructured files $d \in \mathbf{D}$, a set of predefined relationships $r \in \mathbf{R}$, and a set of expected table format examples $\{f_r | r \in \mathbf{R}\}$. The output is a set of structured data in a format of named tables $\mathbf{T} = \{t_r | r \in \mathbf{R}\}$. The data structure for each table t_r is defined by a tuple with pest names and relationship r, such as r =("Pest Name", "Pest Description") for a table of pest descriptions. Extracted entities with relationships are stored in the corresponding table t_r. Algorithm 1 demonstrate the knowledge extraction process.

The three-stage knowledge extraction method includes pre-processing, extraction, and post-processing. In addition to stopword removal and signal removal, an instruction-based

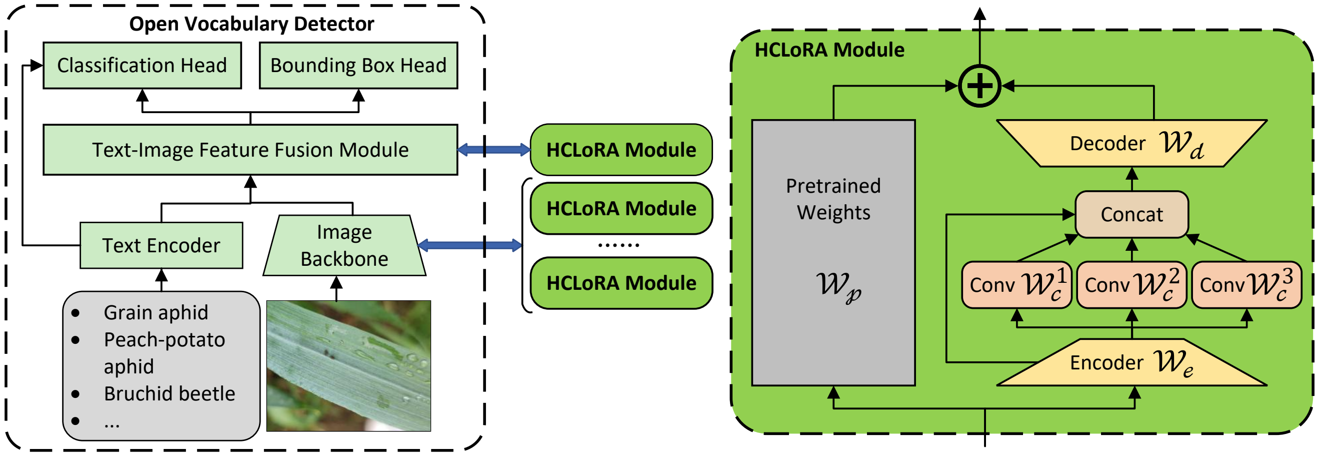


Fig. 4. Open vocabulary detector with HCLoRA modules. OVD is pre-trained on general text-region datasets for zero-shot detection. HCLoRA modules replace some layers in the OVD to efficiently fine-tune the model for pest detection, where Conv 1, Conv 2, and Conv 3 are convolutional layers with different kernel sizes for capturing features with different receptive fields.

Algorithm 1 Algorithm for Knowledge Extraction Method.

Input: A set of unstructured files, $\mathbf{D}$.
A set of predefined relationships, $r \in \mathbf{R}$.
A set of expected table format examples, $\{f_r | r \in \mathbf{R}\}$
Output: A set of named tables, $\mathbf{T}$

$\mathbf{T} \leftarrow \{t_r | r \in \mathbf{R}\}$ ▷ Initialisation
for all $d_i \in \mathbf{D}$ **do**
  $\{p_0, ..., p_k\} \leftarrow split(d_i)$ ▷ Split documents by paragraph
  $\mathbf{C} \leftarrow \{p_0\}$ ▷ Initialisation continuous chunk set
  $idx_o \leftarrow 1$, $idx_n \leftarrow 0$
  while $idx_o \leq k$ **do**
    if $continu(\mathbf{C}[idx_n], p_{idx_o})$ **then**
      $\mathbf{C}[idx_n] \leftarrow \mathbf{C}[idx_n] + p_{idx_o}$ ▷ Merge paragraphs
    else
      $\mathbf{C} \leftarrow \mathbf{C} \cup \{p_{idx_o}\}$ ▷ Add a new paragraph
      $idx_n \leftarrow idx_n + 1$
    end if
    $idx_o \leftarrow idx_o + 1$
  end while
  for all $c \in \mathbf{C}$ **do**
    for all $r \in \mathbf{R}$ **do**
      $k \leftarrow keyExtract(c)$ ▷ Extract keys
      $t_r \leftarrow t_r \cup relExtract(r, f_r, c, k)$ ▷ Construct tables
    end for
  end for
end for
for all $t_r \in T$ **do**
  $t_r \leftarrow merge(t_r)$ ▷ Merge similar entities
  $t_r \leftarrow llmCln(t_r, r)$ ▷ Clean inappropriate entities
end for
return T

document splitting method is utilised to split documents, which avoids relationship loss due to contiguous text splitting. Specifically, the splitting method firstly splits input documents into chunks $\{p_0, ..., p_k\}$ with a fixed length. Then, an instruction learning prompt is employed to detect the continuity of two split chunks for merging semantic continuous chunks. The prompt is formally represented as $continu(p_{idx}, p_{idx+1}) \in \{0, 1\}$ to output a boolean value indicating whether to merge chunks, where p_{idx}, p_{idx+1} refer to the idx chunk and $idx+1$ chunk. The merged chunks are represented as $c \in \mathbf{C}$.

After completing the pre-processing, a heuristic CoT is used to extract knowledge, including pest descriptions, affected crops, pest thresholds, and pest management practices. The first step in CoT is to extract the pest entities as keywords k in a chunk c through $keyExtract(c)$ defined as *"Extracted pest name mentioned in the following text contents, use the vocabulary from the original text. Return a list of pest names without duplicate names. Text Contents: c"*. The second step extracts relational entities to construct named tables t_r by a prompt $relExtract(r, f_r, c, k)$ defined as *"Extracted $\{r\}$ of pest in the PEST LIST mentioned in the following text, use the sentences or words from the original text. Return a list of JSON objects with the pest name as the key of the JSON object and corresponding $\{r\}$ as value, such as $\{f_r\}$. Returns an empty JSON object if no corresponding content is mentioned in the following text. Text Contents: $\{c\}$ PEST LIST: $\{k\}$"*. The output format f_r is a dictionary as an example of output with relationship r for a corresponding table t_r.

Post-processing methods for extracted data are used to merge duplicate data and to remove irrelevant data. Specifically, the tables are traversed based on key values to determine duplicate records. The contents of identified duplicate records are fed into an instruction learning method and are merged by LLMs for updating the entities in the named tables. Finally, the updated entities are passed into LLMs to determine if they are appropriate entities for a corresponding specific relationship. Inappropriate entities are removed from the named table.

D. Suggestion Generation Workflow

To improve the accuracy of suggestions, a suggestion generation method, as shown in Algorithm 2, is proposed based on reasoning, structured data retrieval, and reflection. The reasoning ability of LLMs ensures that multiple environment

Algorithm 2 Algorithm for Suggestion Generation Method.

Input: A user query, q.
Extracted named tables as knowledge bases, $\mathbf{T}$.
Output: A pest management suggestion, sug_{finl}
$k, sug_{cot} \leftarrow cot(q)$ ▷ Generate suggestion proposal
INFO $\leftarrow \{\}$ ▷ Initialisation retrieved information
for all $t_r \in \mathbf{T}$ **do**
  INFO $\leftarrow$ **INFO** $\cup\, retrieval(k, t_r)$ ▷ Retrieve knowledge
end for
$sug_{finl} \leftarrow ref(\mathbf{INFO}, sug_{cot}, q)$ ▷ Optimise suggestion
return sug_{finl}, **INFO**

variables are used sensibly in a suggestion proposal generation process. The retrieval of structured data provides LLMs with external knowledge information. The reflection mechanism combines retrieved information with suggestion generation.

Formally, the first stage is to generate an initial suggestion proposal sug_{cot} for a user query q through a CoT method $cot(q)$, which is implemented by a heuristic prompt *"Let's think step by step."*. In addition, the CoT method extracts keywords in the user query q for information retrieval. The information retrieval is performed through a predefined SQL template within a loop for knowledge bases. Retrieved information, including pest descriptions, pest thresholds, and pest management methods, is spliced into a dictionary as contextual information **INFO** for the suggestion generation. In addition, the sources of retrieved contextual information are recorded to be presented as a basis for decision-making. The last step in the suggestion generation is to generate a final pest management suggestion through the reflection mechanism. Specifically, the reflection mechanism $ref(\mathbf{INFO}, sug_{cot}, q)$ is an instruction-based prompt, defined as *"Acting as an agricultural expert answering user queries in a brief and precise manner based on contextual information and previous suggestions. Context Information: {**INFO**}; Previous Suggestion: {sug_{cot}}; User Query: {q}."*, to combine suggestion proposal sug_{cot} and retrieved information **INFO**. Based on this generation method, users have access to generated management suggestions with related knowledge information.

IV. Experimental Evaluation

The effectiveness of PEZEGO is evaluated in a case study of UK pest management, which covers pest management for carrots, cereals, beans, oilseeds, peas, potatoes, and sugar beet. We compare the performance of proposed methods with state-of-the-art methods on pest detection, information extraction, and suggestion generation tasks in this case study. An ablation experiment is conducted to explore the effect of different LLMs on suggestion generation. In addition, the effectiveness of PEZEGO in agriculture is qualitatively demonstrated through an implementation on Android clients.

A. Experiment Setup

1) Datasets: We extend our pest detection dataset [28], which consists of 9,902 images with annotations provided by agronomists, to validate the effectiveness of HCLoRA. Image samples in this dataset are captured from farms in England. Category annotations contain 27 types of insects. The fine-tuning dataset is randomly sampled from 11 categories in this dataset. The remaining samples are used for validation.

A pest management encyclopaedia [29] from the Agriculture and Horticulture Development Board is used to validate the knowledge extraction method. A set of manually extracted data is recorded as ground truth values of the extraction for pest types, affected crops, pest descriptions, pest thresholds, and management practices. The data samples are long text descriptions, except for pest types and affected crops.

To evaluate suggestion generation methods, a tabular dataset of pest sampling records is synthesised for simulating user queries. Specifically, the tabular dataset contains pest statistic information, crop type, crop growth stage, time of collection, location, weather, and temperature. Pest statistic information records observed pest species and their corresponding population densities, such as pollen beetle with 5 adults per square meter. The crop type and growth stage describe the crop information, such as the oilseed on flower-bud emergence stage (GS 55). The time of collection is recorded as the day of the year. Based on the aforementioned information in pest sampling records, a rule-based expert system [30] driven by agricultural expert knowledge is employed to annotate pest sampling records with ground truth values of whether or not management is required. In addition, tabular data is inserted into pre-defined text templates to construct user queries.

2) Evaluation Metrics: Mean average precision (AP) is utilised to evaluate the detection ability, which is defined as an average area under precision-recall curves with an intersection over union threshold of 0.5 for all categories in the validation dataset. To demonstrate the effectiveness of fine-tuning methods, AP for unseen categories AP^u and seen categories AP^s is calculated. The seen and unseen categories are defined as pest species present or absent in the fine-tuned training set, respectively. To comprehensively validate the performance of fine-tuned models for seen and unseen categories, a harmonised AP^h is defined as,

$$AP^h = \frac{2 \times AP^u \times AP^s}{AP^u + AP^s}. \tag{3}$$

In addition to detection abilities, model size and detection speed are reported to demonstrate the availability of OVDs in a pest detection task.

For knowledge extraction and suggestion generation, evaluation metrics in machine learning, including accuracy, precision, and F1 score, are employed to evaluate the effectiveness of proposed methods. Varied definitions of true positive samples are provided for two tasks. Specifically, true positive samples of the entities with precise vocabularies, such as pest types and affected crops, are defined as samples that are identical to ground-truth samples in the knowledge extraction task. The entities of long textual descriptions, such as pest descriptions and management practice, utilise BERTScore [31] to measure the textual similarity between extracted values and ground-truth values. BERTScore is implemented in this work using the 384-dimensional embedding vector space of the MiniLM-L6-v2 model with cosine similarity. A threshold

TABLE I
PERFORMANCES OF OVDS AND FINE-TUNING METHODS. YOLOWORLD ACHIEVES THE HIGHEST DETECTION PERFORMANCE IN PEST DETECTION TASKS. FINE-TUNED YOLOWORLD ACHIEVED THE HIGHEST AP^h THROUGH HCLORA.

Models	Tuning Strategy	Model size (MB)	Speed (FPS)	AP	AP^h	AP^s	AP^u
GLIP-T	-	2,317.8	4.2	0.0043	0.0008	0.0098	0.0004
OWLViT	-	584.6	11.9	0.0097	0.0084	0.0154	0.0058
OmDet	-	440.43	12.3	0.0019	0.0003	0.0044	0.0002
Grounding DINO-T	-	661.8	2.87	0.0026	0.0021	0.0043	0.0014
YOLOWorld	-	24.7	136.9	**0.0238**	**0.0388**	**0.0330**	**0.0470**
YOLOWorld	FFT	24.6	144.9	0.284	0.0698	**0.6437**	0.0369
YOLOWorld	SPEFT	24.6	153.8	0.284	0.0698	0.6435	0.0369
YOLOWorld	PFT	24.6	148.7	0.032	0.0128	0.0696	0.0157
YOLOWorld	LoRA	24.8	157.9	0.163	0.0752	0.3378	0.0423
YOLOWorld	HCLoRA	36.9	151.3	**0.297**	**0.1759**	0.5792	**0.1037**

for BERTScore is set to 0.5 to distinguish between positive and negative samples. Entity-level F1 (E F1) is [32] is used to report the performance of knowledge extraction methods.

For the suggestion generation method, the record information in the dataset is inserted into a set of pre-defined text templates as user queries for pest management suggestion generation. A classification instruction of LLMs is used to assign labels for the suggestions generated by different methods. Positive samples of generated suggestions are defined as suggestions that require immediate management through voting from various LLMs. Negative samples are the opposite. The samples that do not provide a clear decision in generated suggestions are defined as unknown samples.

3) Baselines: The performances of OVDs, including GLIP-T [33], OWLViT [34], OmDet [35], Grounding DINO-T [36], YOLOWorld [9] for pest detection, are validated to determine the optimal baseline model for fine-tuning methods. Advanced fine-tuning methods, including full-parameter (FFT) [37], parameter-efficient (SPEFT) [38], prompt (PFT) [39], and LoRA [27] fine-tuning, are compared with HCLoRA to demonstrate its validity.

The proposed knowledge extraction method is compared with the state-of-the-art methods including in-context learning [16], Resolved GPT-3 (GPT-3+R) [40], Evaporate-Direct (Evapo-D), Evaporate-Code (Evapo-C) [32], and Self-Verification (Self-verif) [41] for knowledge extraction. The in-context learning method feeds a task instruction with two input-output pairs into LLMs for guiding information extraction. GPT-3+R and Evapo-D are zero-shot instruction learning methods with different task instructions and post-processing methods. Evapo-C uses four samples to generate Python code that extracts information through regular expressions, reducing the computational cost of LLMs. Self-verif revises the extracted information for improving extraction performance.

The baselines of suggestion generation include instruction learning [15], Auto-CoT [42], tree-of-through (ToT) [43], Least-to-Most [44], Self-Ask [45], and RAG [46]. These baselines cover methods to improve generation performance through appending reasoning and information retrieval capabilities. Auto-CoT and ToT inspire the reasoning of LLMs to improve the accuracy of suggestions. Least-to-Most and Self-Ask identify a set of subproblems for answering user queries in a zero-shot manner and a few-shot manner, respectively. RAG retrieves text based on embedding vectors of documents to optimise generation.

4) Experiment and System Implementation: Fine-tuning and validation of OVDs are completed on a server with 3060 GPU. Langchain toolkit [47] is used to implement LLM-based workflows. All of the LLM-based workflows in the system use the GPT-3.5-turbo (GPT-3.5) [48] as a basis of LLMs. The mobile application of the PEZEGO system is implemented by Kotlin with Jetpack. API services, which are deployed on a cloud computing cluster, are coded by Python and Java for different API services.

B. Comparison With Existing Work

Table I demonstrates the performance of OVDs and fine-tuning methods on pest detection. Compared with other state-of-the-art OVDs, YOLOWorld [9] achieves the highest detection speed and average precision for zero-shot pest detection due to its efficient one-stage model structure with convolution. Therefore, fine-tuning methods are validated on YOLOWorld.

In the validation of fine-tuning methods, FFT achieves the highest AP^s due to adjusting all trainable parameters of YOLOWorld for adapting the pest detection task, while losing the ability to detect unseen categories due to the lack of constraints on the pre-training weights. Similarly, the SPEFT method that optimises classification and bounding box heads loses the ability to detect unseen categories. Counterintuitively, PFT obtains the lowest performances for both seen categories AP^s and unseen categories AP^u of all the fine-tuning due to few trainable parameters. Moreover, the adjusted text embeddings affect the detection of unseen categories. In contrast, LoRA obtains a balanced detection ability for seen and unseen categories by parallel residual connections. Compared with LoRA, the proposed HCLoRA achieves higher AP^s and AP^u with separating features processing and multiple receptive field fusions. In addition, HCLoRA learns generalisable pest features by low-rank constraints, enabling positive enhancement of fine-tuning for unseen pest categories, rather than just maintaining the existing zero-shot detection capability of OVDs. For example, learning visual features of grain aphids has a positive impact on the detection of willow-carrot aphids.

The performances of baselines and the proposed workflow on knowledge extraction tasks are shown in Fig. 5. Different knowledge extraction methods have significant differences in performance when dealing with different types of data. The proposed knowledge extraction method obtained the best performance with 68.70% average E F1 due to the LLM-based pre-processing and post-processing. In particular, this

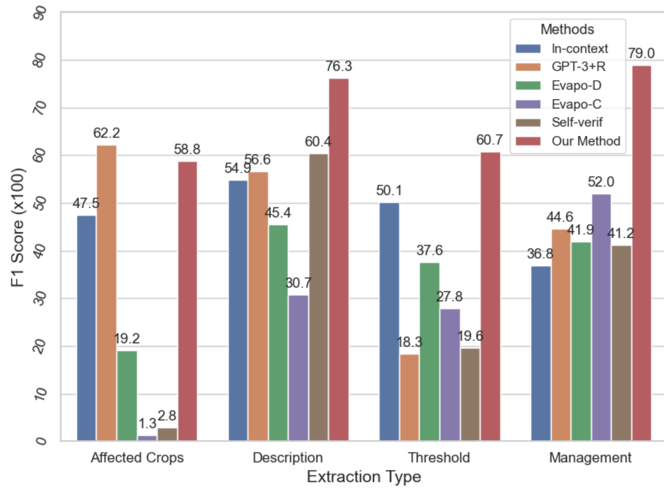


Fig. 5. Knowledge extraction method performances for affected crops, description of pests, economic thresholds, and management strategies compared with in-context learning (In-context), GPT-3+R, Evapo-D, Evapo-C, and Self-verif on Entity-level F1 value (E F1).

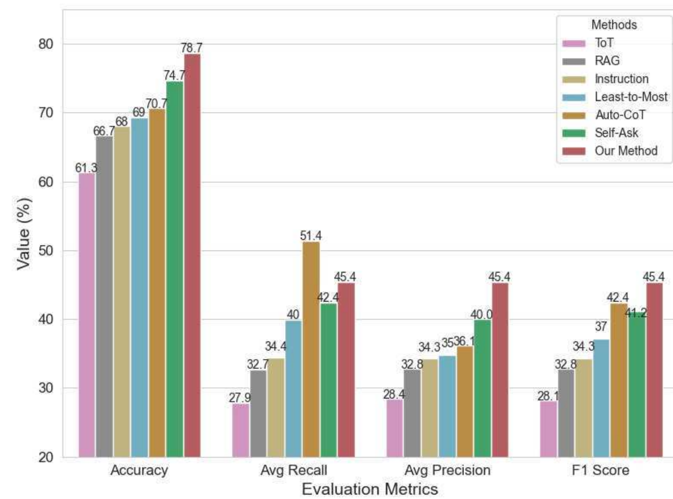


Fig. 6. Suggestion generation workflow performance compared with Instruction Learning (Instruction), Auto-CoT, ToT, Least-to-Most, Self-Ask, and RAG on average Accuracy, average Recall, average Precision, and F1 score.

workflow outperforms other baseline methods in the extraction of pest descriptions, pest thresholds, and pest management. However, the proposed workflow slightly lags behind GPT-3+R in the extraction of affected crop information, which is explicit words rather than long text. Since GPT-3+R uses regular expression matching, it avoids the word conversion of LLM and is more adept at extracting words. In addition, few-shot learning methods, such as in-context learning and Evapo-C, have not achieved satisfactory accuracy in extraction tasks. Because few samples do not demonstrate an adequate input-output relationship for supporting information extraction. Especially, Evapo-C, which generate regular expression for extraction, is difficult to extract entities from complex inputs. Self-Verify utilises the multiple generation of LLMs to optimise the extraction results, which also results in the modification of the specific entity vocabulary, affecting the accuracy of the entity extraction.

For the evaluation of suggestion generation, the proposed generation method outperforms other baseline methods on accuracy. The proposed method improves the average accuracy by nearly 15% compared to RAG, which accesses the original textual information by vector-based retrieval, since the structured knowledge retrieval method is more accurate for complex generative tasks. Self-Ask using an in-context learning method with four samples obtains the suboptimal accuracy. However, Self-Ask is limited by the reasoning process for the suggestion generation task. Throughout the evaluation of the suggestion generation task, the low average precision and average recall value are due to the absence of samples with unknown ground truth values in the test dataset. Specifically, there are some generated suggestions from LLMs where an explicit decision is not provided to avoid responsibility for decision-making, even if the need for a decision is declared in the task instruction. Therefore, there are no true positive samples for the samples without explicit decisions.

TABLE II
SUGGESTION GENERATION TASK PERFORMANCES FOR DIFFERENT LLMS INCLUDING FLAN, GPT-4, GPT-3.5, GPT-3 AND GPT-BASE.THE HIGHEST PERFORMANCE OBTAINED BY THE FIVE LLMS IS BOLDED.

Methods	Acc	Avg Prec	Avg Rec	F1 Score
Flan	62.67%	28.95%	28.59%	28.77%
GPT-4	73.33%	39.00%	**46.72%**	42.51%
GPT-3.5	**78.72%**	**45.40%**	45.40%	**45.40%**
GPT-3	60.00%	41.66%	41.66%	41.66%
GPT-Base	54.67%	20.83%	19.00%	19.87%

C. Ablation Experiment

The impact of using different LLMs on the performance of suggestion generation service is demonstrated in Table II. The GPT-3.5 model achieves the highest average accuracy, precision, and F1 scores on suggestion generation for pest management. In addition, the GPT-4 model only obtains the highest average recall. GPT-4 does not obtain the highest performance because it is overconfident and ignores the provided contextual information for generation. These two models have been trained on downstream tasks for instruction learning and are therefore more adept at understanding task descriptions and completing tasks. Flan, GPT-3 and GPT-based models do not obtain better performance due to the limitation of training strategies and training data. This result demonstrates that the performance of the proposed framework depends on the zero-shot and few-shot learning capability of LLMs.

D. Qualitative Analysis For Mobile Application

A suggestion example generated by PEZEGO and the user interfaces of PEZEGO are demonstrated in Fig. 7. As shown on the home page, users can check the pest detection records and suggestion records by the mobile application. The pest detection function reduces the knowledge requirements for farmers on pest identification. Environmental information is collected by the mobile application for reliable suggestion generation. While the original LLM and Auto-CoT method

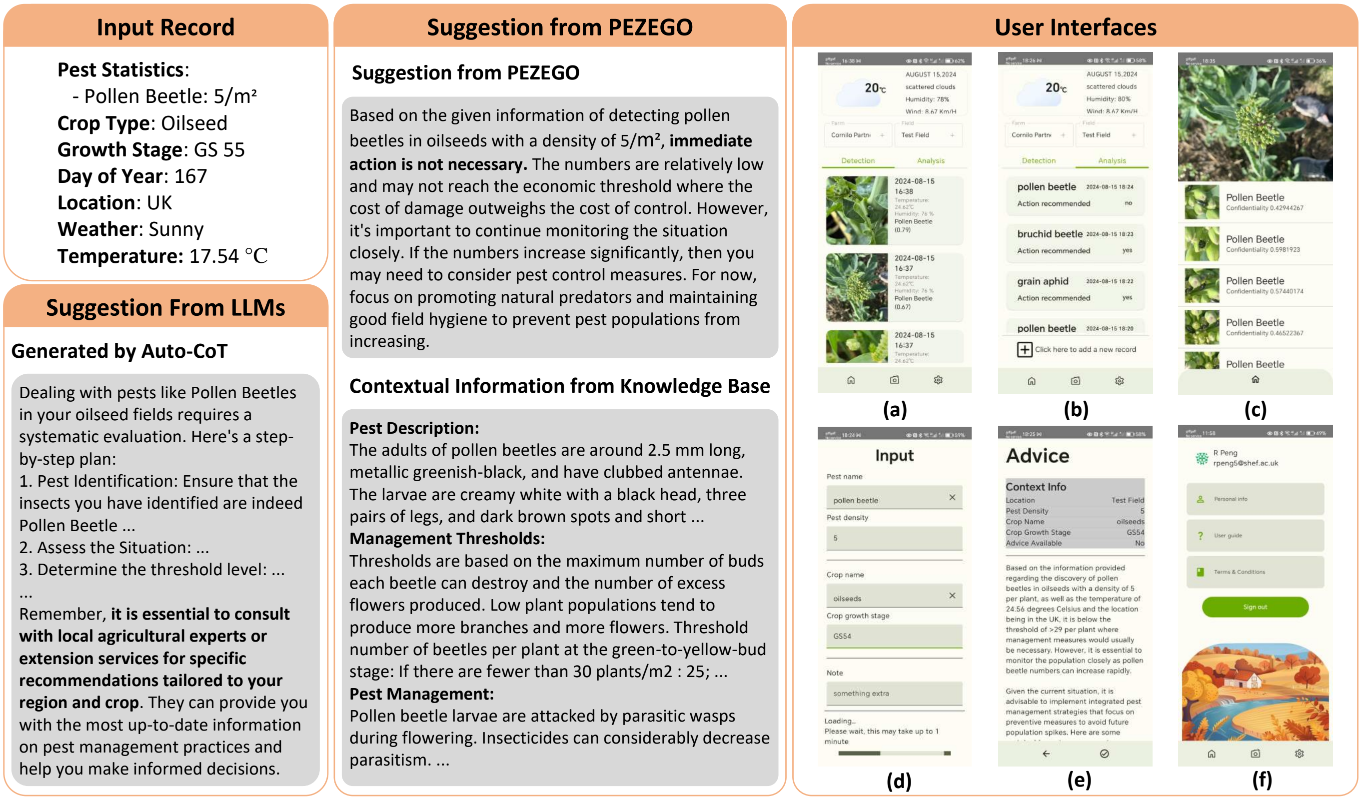


Fig. 7. Suggestion examples and user interface of PEZEGO. The content in the input record is a record from the test dataset. Compared to Auto-CoT, PEZEGO provides a clearer suggestion with domain knowledge. The user interface includes (a) Home Page for Pest Detection, (b) Home Page for Suggestions, (c) Pest Detection Page, (d) Suggestion Generation Page, (e) Suggestion Screen, and (f) Setting Screen.

has the ability to provide users with pest management suggestions, this method tends to generate unclear management decisions with disclaimers to avoid liability. In comparison, the PEZEGO system generates clear suggestions for decision-making, accompanied by relevant knowledge information with economic threshold and non-chemical management methods.

V. Discussion and Further Research Direction

Although the above case studies qualitatively and quantitatively demonstrate the effectiveness of PEZEGO in supporting sustainable agriculture practices by suggestion generation, this work still has some limitations, which lead to further research directions. Firstly, zero-shot pest detection ability for OVDs is limited due to the differences between the pre-training and pest image data. Incremental pre-training methods with more diverse pest samples need to be further explored for further improving zero-shot pest detection ability. Secondly, there is still a trade-off between accuracy and effectiveness for knowledge base construction methods. The effectiveness of structured knowledge bases for suggestion generation has been shown in experiments. However, the accuracy of extracting structured data through automated methods remains limited. Therefore, an accurate and effective solution needs to be explored. Thirdly, the validation of this work focuses on decision-making performance based on LLMs and environmental information. However, LLMs have the potential to enable open question-and-answer for supporting agricultural management. In addition, the effectiveness of management strategies still needs to be verified. Therefore, a more extensive validation for agricultural management needs to be completed, covering open question-and-answer aand managment strategies. Finally, the PEZEGO system provides data storage but does not address the cybersecurity threats in the IoT environment. Developing a robust security framework that includes advanced encryption, intrusion detection and strict access control measures to protect the privacy of sensitive agricultural data is a further direction.

VI. Conclusion

Precision agriculture systems aim to implement sustainable agriculture management practices, which is hindered by the detection of numerous pest species, lack of structured agricultural knowledge, and precision suggestion generation with reasoning. In this work, an IoT-based precision agriculture system, called PEZEGO, is proposed to address the aforementioned challenges through LLMs with environmental information. Specifically, the feasibility of OVDs for zero-shot pest detection is explored by fine-tuning. The proposed fine-tuning method, HCLoRA, significantly improves the performance of OVDs for pest detection, which achieves 0.297 AP for 27 categories, 0.5792 AP^s for 11 seen categories, 0.1037 AP^u for 16 unseen categories, and 151.3 FPS. In addition, a SRAG workflow, consisting of agricultural knowledge extraction and management suggestion generation methods, is proposed to provide accurate management suggestions. In practice, the knowledge extraction method utilises LLMs to structure agriculture knowledge, including pests, crops, thresholds, and management strategies, as external resources for supporting precision suggestion generation. To address the hallucination issue in generation by LLMs, we propose an optimised RAG workflow, which integrates with reasoning, structured data retrieval, and reflection mechanisms to enhance the accuracy of suggestion generation. In quantitative experiments, we validate the effectiveness of the proposed methods using a pest encyclopaedia dataset and a pest sampling dataset. Compared to state-of-the-art zero-shot and few-shot methods, the proposed methods achieve optimal performance in the knowledge extraction tasks, including pest description, pest thresholds, and pest management practices. Furthermore, the PEZEGO achieves state-of-the-art results on three evaluation metrics in the generation task for pest management suggestions with 77.33% average accuracy. In ablation experiments for the choice of LLMs, we note that GPT-3.5 achieves optimal accuracy in suggestion generation. GPT-4 does not demonstrate the expected performance due to overconfidence issues. In addition, the client of PEZEGO is implemented on an Android mobile application, which provides relevant knowledge sources with generated suggestions to ensure reliability.

References

[1] G. S. Malhi, M. Kaur, and P. Kaushik, "Impact of climate change on agriculture and its mitigation strategies: A review," *Sustainability*, vol. 13, no. 3, p. 1318, 2021.

[2] S. Savary, L. Willocquet, S. J. Pethybridge, P. Esker, N. McRoberts, and A. Nelson, "The global burden of pathogens and pests on major food crops," *Nature ecology & evolution*, vol. 3, no. 3, pp. 430–439, 2019.

[3] M. Ramsden, S. Kendall, S. Ellis, and P. Berry, "A review of economic thresholds for invertebrate pests in uk arable crops," *Crop protection*, vol. 96, pp. 30–43, 2017.

[4] Z. Yuan, N. Musa, K. Dybal, M. Back, D. Leybourne, and P. Yang, "Quantifying nematodes through images: Datasets, models, and baselines of deep learning," in *2023 IEEE 22nd International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. IEEE, 2023, pp. 2448–2455.

[5] G. Garg, S. Gupta, P. Mishra, A. Vidyarthi, A. Singh, and A. Ali, "Cropcare: an intelligent real-time sustainable iot system for crop disease detection using mobile vision," *IEEE Internet of Things Journal*, vol. 10, no. 4, pp. 2840–2851, 2021.

[6] H. Chen, Y. Lan, B. K. Fritz, W. C. Hoffmann, and S. Liu, "Review of agricultural spraying technologies for plant protection using unmanned aerial vehicle (uav)," *International Journal of Agricultural and Biological Engineering*, vol. 14, no. 1, pp. 38–49, 2021.

[7] R. P. Sharma, D. Ramesh, P. Pal, S. Tripathi, and C. Kumar, "Iot-enabled ieee 802.15. 4 wsn monitoring infrastructure-driven fuzzy-logic-based crop pest prediction," *IEEE Internet of Things Journal*, vol. 9, no. 4, pp. 3037–3045, 2021.

[8] N. Desneux, P. Han, R. Mansour, J. Arnó, T. Brévault, M. R. Campos, A. Chailleux, R. N. Guedes, J. Karimi, K. A. J. Konan *et al.*, "Integrated pest management of tuta absoluta: practical implementations across different world regions," *Journal of Pest Science*, pp. 1–23, 2022.

[9] T. Cheng, L. Song, Y. Ge, W. Liu, X. Wang, and Y. Shan, "Yolo-world: Real-time open-vocabulary object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 16 901–16 911.

[10] B. Min, H. Ross, E. Sulem, A. P. B. Veyseh, T. H. Nguyen, O. Sainz, E. Agirre, I. Heintz, and D. Roth, "Recent advances in natural language processing via large pre-trained language models: A survey," *ACM Computing Surveys*, 2021.

[11] W. Zhang, H. Huang, Y. Sun, and X. Wu, "Agripest-yolo: A rapid light-trap agricultural pest detection method based on deep learning," *Frontiers in Plant Science*, vol. 13, p. 1079384, 2022.

[12] C.-J. Chen, Y.-Y. Huang, Y.-S. Li, Y.-C. Chen, C.-Y. Chang, and Y.-M. Huang, "Identification of fruit tree pests with deep learning on embedded drone to achieve accurate pesticide spraying," *IEEE Access*, vol. 9, pp. 21 986–21 997, 2021.

[13] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[14] P. Gardner, R. Fuentes, N. Dervilis, C. Mineo, S. Pierce, E. Cross, and K. Worden, "Machine learning at the interface of structural health monitoring and non-destructive evaluation," *Philosophical Transactions of the Royal Society A*, vol. 378, no. 2182, p. 20190581, 2020.

[15] R. Lou, K. Zhang, and W. Yin, "Large language model instruction following: A survey of progresses and challenges," *Computational Linguistics*, vol. 50, no. 3, pp. 1053–1095, 2024.

[16] R. Agarwal, A. Singh, L. Zhang, B. Bohnet, L. Rosias, S. Chan, B. Zhang, A. Anand, Z. Abbas, A. Nova *et al.*, "Many-shot in-context learning," *Advances in Neural Information Processing Systems*, vol. 37, pp. 76 930–76 966, 2024.

[17] A. Saparov and H. He, "Language models are greedy reasoners: A systematic formal analysis of chain-of-thought," 2023.

[18] Z. Wan, F. Cheng, Z. Mao, Q. Liu, H. Song, J. Li, and S. Kurohashi, "Gpt-re: In-context learning for relation extraction using large language models," 2023.

[19] S. Sia and K. Duh, "In-context learning as maintaining coherency: A study of on-the-fly machine translation using large language models," pp. 173–185, 2023.

[20] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou *et al.*, "Chain-of-thought prompting elicits reasoning in large language models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 24 824–24 837, 2022.

[21] M. Huh, B. Cheung, T. Wang, and P. Isola, "Position: The platonic representation hypothesis," in *Forty-first International Conference on Machine Learning*, 2024.

[22] J. Li, Z. Guan, J. Wang, C. Y. Cheung, Y. Zheng, L.-L. Lim, C. C. Lim, P. Ruamviboonsuk, R. Raman, L. Corsino *et al.*, "Integrated image-based deep learning and language models for primary diabetes care," *Nature medicine*, pp. 1–11, 2024.

[23] D. Shu, H. Zhao, X. Liu, D. Demeter, M. Du, and Y. Zhang, "Lawllm: Law large language model for the us legal system," in *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, 2024, pp. 4882–4889.

[24] X. Liu, Z. Liu, H. Hu, Z. Chen, K. Wang, K. Wang, and S. Lian, "A multimodal benchmark dataset and model for crop disease diagnosis," in *European Conference on Computer Vision*. Springer, 2024, pp. 157–170.

[25] S. Yang, Z. Yuan, S. Li, R. Peng, K. Liu, and P. Yang, "Advancing agricultural decision-making with a multi-dimensional evaluation of large language models for sustainable pest management," in *2024 IEEE 22nd International Conference on Industrial Informatics (INDIN)*. IEEE, 2024, pp. 1–10.

[26] J. Qing, X. Deng, Y. Lan, and Z. Li, "Gpt-aided diagnosis on agricultural image based on a new light yolopc," *Computers and electronics in agriculture*, vol. 213, p. 108168, 2023.

[27] 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," in *International Conference on Learning Representations*, 2022. [Online]. Available: https://openreview.net/forum?id=nZeVKeeFYf9

[28] Z. Yuan, S. Li, R. Peng, D. Leybourne, P. Yang, and Y. Li, "Pestdss: An integrated decision support system for sustainable pest management in agriculture," in *2023 IEEE 32nd International Symposium on Industrial Electronics (ISIE)*. IEEE, 2023, pp. 1–6.

[29] S. Ellis, S. White, J. Holland, B. Smith, R. Collier, and A. Jukes, "Encyclopaedia of pests and natural enemies in field crops," 2014.

[30] S. Li, Z. Yuan, R. Peng, D. Leybourne, Q. Xue, Y. Li, and P. Yang, "An effective farmer-centred mobile intelligence solution using lightweight deep learning for integrated wheat pest management," *Journal of Industrial Information Integration*, vol. 42, p. 100705, 2024.

[31] A. R. Fabbri, W. Kryściński, B. McCann, C. Xiong, R. Socher, and D. Radev, "Summeval: Re-evaluating summarization evaluation," *Transactions of the Association for Computational Linguistics*, vol. 9, pp. 391–409, 2021.

[32] S. Arora, B. Yang, S. Eyuboglu, A. Narayan, A. Hojel, I. Trummer, and C. Ré, "Language models enable simple systems for generating structured views of heterogeneous data lakes," *Proceedings of the VLDB Endowment*, vol. 17, no. 2, pp. 92–105, 2023.

[33] L. H. Li, P. Zhang, H. Zhang, J. Yang, C. Li, Y. Zhong, L. Wang, L. Yuan, L. Zhang, J.-N. Hwang *et al.*, "Grounded language-image pre-training," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 965–10 975.

[34] M. Minderer, A. Gritsenko, A. Stone, M. Neumann, D. Weissenborn, A. Dosovitskiy, A. Mahendran, A. Arnab, M. Dehghani, Z. Shen *et al.*, "Simple open-vocabulary object detection," in *European Conference on Computer Vision*. Springer, 2022, pp. 728–755.

[35] T. Zhao, P. Liu, and K. Lee, "Omdet: Large-scale vision-language multi-dataset pre-training with multimodal detection network," *IET Computer Vision*, 2024.

[36] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, Q. Jiang, C. Li, J. Yang, H. Su *et al.*, "Grounding dino: Marrying dino with grounded pre-training for open-set object detection," in *European Conference on Computer Vision*. Springer, 2025, pp. 38–55.

[37] N. Tajbakhsh, J. Y. Shin, S. R. Gurudu, R. T. Hurst, C. B. Kendall, M. B. Gotway, and J. Liang, "Convolutional neural networks for medical image analysis: Full training or fine tuning?" *IEEE transactions on medical imaging*, vol. 35, no. 5, pp. 1299–1312, 2016.

[38] Z. Fu, H. Yang, A. M.-C. So, W. Lam, L. Bing, and N. Collier, "On the effectiveness of parameter-efficient fine-tuning," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 37, no. 11, 2023, pp. 12 799–12 807.

[39] B. Lester, R. Al-Rfou, and N. Constant, "The power of scale for parameter-efficient prompt tuning," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021, pp. 3045–3059.

[40] M. Agrawal, S. Hegselmann, H. Lang, Y. Kim, and D. Sontag, "Large language models are few-shot clinical information extractors," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 2022, pp. 1998–2022.

[41] Z. Gero, C. Singh, H. Cheng, T. Naumann, M. Galley, J. Gao, and H. Poon, "Self-verification improves few-shot clinical information extraction," 2023.

[42] Z. Zhang, A. Zhang, M. Li, and A. Smola, "Automatic chain of thought prompting in large language models," 2023.

[43] S. Yao, D. Yu, J. Zhao, I. Shafran, T. Griffiths, Y. Cao, and K. Narasimhan, "Tree of thoughts: Deliberate problem solving with large language models," *Advances in neural information processing systems*, vol. 36, pp. 11 809–11 822, 2023.

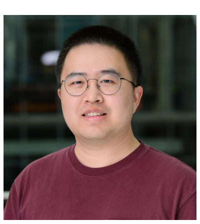
[44] D. Zhou, N. Schärli, L. Hou, J. Wei, N. Scales, X. Wang, D. Schuurmans, C. Cui, O. Bousquet, Q. Le *et al.*, "Least-to-most prompting enables complex reasoning in large language models," 2023.

[45] O. Press, M. Zhang, S. Min, L. Schmidt, N. A. Smith, and M. Lewis, "Measuring and narrowing the compositionality gap in language models," pp. 5687–5711, 2023.

[46] Y. Wang, X. Ma, and W. Chen, "Augmenting black-box llms with medical textbooks for biomedical question answering," pp. 1754–1770, 2024.

[47] O. Topsakal and T. C. Akinci, "Creating large language model applications utilizing langchain: A primer on developing llm apps fast," in *International Conference on Applied Engineering and Natural Sciences*, vol. 1, no. 1, 2023, pp. 1050–1056.

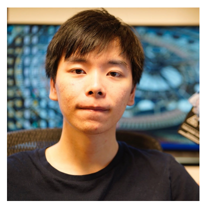
[48] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.



Zhipeng Yuan (Student Member, IEEE) received a BSc from Northeastern University, China, in 2018, and an MSc in Computer Science from the University of Sheffield, UK, in 2020. He is working toward a PhD in the Department of Computer Sciences, the University of Sheffield, UK. He has published over 10 research articles. His research fields include deep learning, explainability, and application machine learning in agriculture, industry, and healthcare. He has been serving as a reviewer for top/leading international journals and conferences.

Kang Liu (Member, IEEE) received the Ph.D. degree in control science and engineering from the University of Science and Technology of China, Hefei, China, in 2022. He is currently a Postdoctoral Fellow with the Department of Aeronautical and Aviation Engineering, Hong Kong Polytechnic University, Hong Kong. Before he was with the Hong Kong Polytechnic University, he was a Research Associate with the University of Sheffield, UK. His research interests include theoretical researches and engineering applications in intelligence control and precision agriculture, such as robot dynamics, learning/fuzzy-based control, unmanned aerial vehicles, autonomous vehicles, drone remote sensing, as well as pest and disease identification, He has been serving as a reviewer for many top/leading international journals and conferences. He is a Member of IEEE/RAS Technical Committee on Multi-Robot Systems, Aerial Robotics and Unmanned Aerial Vehicles, and Agricultural Robotics and Automation.

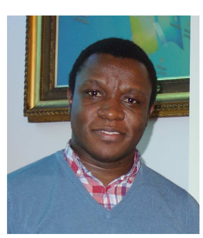
Shunbao Li was awarded the B.Sc. degree in Automation from Yanshan University, China, in 2017, the M.Sc. degree in Software Systems and Internet Technology from the University of Sheffield, in 2019. Currently, he is a PhD Student of computer science at the University of Sheffield. His research interests include Multi-modality data fusion and Deep learning.



Ruoling Peng is a Research Assistant and PhD researcher in the Computer Science department, the University of Sheffield. He graduated from the EEE department of the University of Sheffield with a BEng degree and later received an MRes in EEE from the University of Sheffield. He is interested in agent-based epidemic simulation with mobility.

Po Yang (Senior Member, IEEE) is a Senior Lecturer of Large-scale Data Fusion in the Computer Science Department (COM) at the University of Sheffield (TUoS) and also the Deputy Director of Research and Innovation in COM. Po has extensive research experience in pervasive and mobile intelligence, where he conducted interdisciplinary research across machine learning, pervasive computing, smart sensing, and data science. He has been the Project Coordinator on over £3.2 million of research funding from InnovateUK, BBSRC, STFC, EPSRC, Research England, industrial funds, etc. He has published 100+ journal papers and 80+ IEEE/ACM conference papers. He serves as an Associate Editor in IEEE Journal of Translational Engineering in Health and Medicine, Journal of Biomedical Informatics and Scientific Reports.

Daniel Leybourne (Fellow of the Royal Entomological Society) is a Research Fellow at the University of Liverpool. He is funded by the Royal Commission or the Exhibition of 1851. Broadly, his research aim is to understand the biological and ecological processes that underpin the success of herbivorous insects and insect-vectored viruses in agricultural systems.



Nasamu Musa is an early career pest management researcher with experience and training for over a decade. Within ADAS, he conducts work on projects that involves managing arable crop pests. This includes revising pest thresholds, understanding insecticide resistance development, and supporting the development of new technologies such as AI tools for pest identification. He also manages the Pest Evaluation Services and Pest Biology Laboratory at ADAS High Mowthorpe.