scientific reports

Check for updates

OPEN

Smart customer service in unmanned retail store enhanced by large language model

Wang Wang[1], Ping Zhang[2]✉, Changxia Sun[3] & Dengchao Feng[4]

In unmanned retail store, providing smart customer service requires two stages: understanding customer needs, and guiding the customer to the product. In this paper, we propose an end-to-end (Customer-to-Shelf) software service framework for unmanned retail. The framework integrates visual recognition technology to detect retail objects, large language models to analyze customer shopping needs and make proper recommendations. First, deep neural network based image recognition models are studied for implementing effective stock keeping units (SKUs) object recognition on the shelf. Second, a novel method is proposed to fine-tune large language models (LLMs) with limited training dataset. Metaheuristic approaches are used to optimize the mask locations in a low dimensional parameter space, resulting a more efficient parameter updating method for limited downstream data. Third, by facilitating an automatic analysis of customer preferences powered by large language models, we present a smart recommender system based on domain-specific knowledge, which completes the Customer-to-Shelf software service framework. Experimental results show that our proposed fine-tuning method, is more efficient than other state-of-the-art training methods for limited downstream domain dataset. Using fine-tuned large models, we can successfully create a seamless shopping experience for customers by understanding personalized needs and providing shopping advice in the unmanned retail store.

With the rapid development of intelligent technology, the traditional retail industry is undergoing a transformation towards digitization and intelligence[1]. Unmanned retail is currently a hot topic in the field of AI applications[2–6]. It encompasses theoretical challenges in AI and holds significant practical value. The lives of many urban residents are inseparable from smart retail.

The work on stock keeping units (SKUs) product recognition has garnered widespread research attention. Examples of retail SKUs on shelves, from SKU110k[7] and RPC datasets, are given in Fig. 1. Automatic recognition of these retail goods may greatly improve the operational efficiency. Geng et al.[8] proposed to study fine-grained recognition of SKUs. Their research work is distinctive in that it enables the training and shelving of products to be accomplished through a single image. Wang et al.[9] also studied the retail object training in one shot. They proposed to use Siamese network structure to realize the model training. Wang et al.[10] proposed to use modified residual network structure to improve the accuracy of retail object recognition. They designed a boundary regression method especially for densely placed objects in retail shelves. The method Wei et al.[11] proposed a publicly available retail product dataset and conducted research on Automatic Check-Out. They employed a coarse-to-fine approach and a dual pyramid scale network to enhance the effectiveness of retail product recognition. However, in their study, the contribution to unmanned retail was limited to commodity recognition, with insufficient discussion on the operation of retail stores. Other processes such as product recommendation and shopping guidance can also benefit from AI technology. Santra et al.[12] proposed a model that can distinguish fine-grained differences. In the model, part-level cues were encoded using convolutional LSTM(Long Short-Term Memory). Similarly, such recognition capability, when combined with product recommendations, would further enhance the service quality of unmanned retail. Dworakowski et al.[13] proposed a robotic architecture designed to assist customers in locating retail goods. Their work presents a challenging task that could prove beneficial for end-to-end services by incorporating an understanding of customer needs and intelligent recommendations.

[1]College of Information and Digital Engineering, Luoyang Vocational College of Science and Technology, Luoyang 471023, Henan, China. [2]College of Mathematics and Statistics, Henan University of Science and Technology, Luoyang 471023, Henan, China. [3]College of Information and Management Sciences, Henan Agricultural University, Zhengzhou 450046, Henan, China. [4]Police Drone Warfare Training and Research Center, Shandong Police College, Jinan 250200, Shandong, China. ✉email: zping@haust.edu.cn

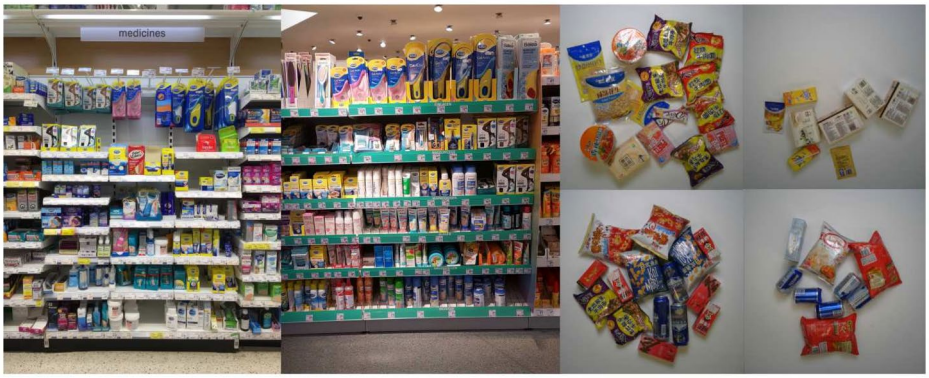


Fig. 1. Depiction of retail SKUs on store shelves: left are from SKU110k and the right are from RPC dataset.

In visual object recognition, the challenges posed by lighting conditions, occlusions, multiple angles, and distances in the retail store environment present difficulties for visual target recognition. The recognition technology for visual targets has undergone rapid development. Early methods focused on addressing occlusion and lighting effects by employing more robust image features[14–16]. Serre et al.[14] investigated target recognition, particularly suitable for complex visual scenes. The proposed features, through maximum pooling operations, also exhibited robustness. However, when considering retail products, there was significant uniqueness, with only minor differences in the appearance of different items.

These conventional non-deep learning methods had advantages in hand-crafted feature engineering, while deep networks excelled in representation learning, given a sufficient amount of data. However, deep network approaches also have drawbacks, notably high computational resource consumption, particularly in unmanned retail scenarios where training and recognizing retail SKU models require lower resources and faster speeds. He et al.[17] invented the residual neural network architecture. Residual network structures are often used as the backbone network in object recognition networks. The You Only Look Once (YOLO) series[18,19] have demonstrated excellent performance in object recognition, with significant advantages in both accuracy and computational efficiency[20,21]. Leo et al.[22] studied using convolutional network architecture for retail object recognition. According to these findings, the CNN architecture is the state-of-the-art approach to SKUs recognition.

In our smart retail research, we further study the recommendation of SKUs. Smart recommender applications can harness the power of language understanding, where the large language models have become increasingly popular in recent years. In many of the current popular applications of large language models, the focus often revolves around chat and interaction, with topics predominantly drawn from widely accessible internet discussions. However, the application of large language models in specific industrial scenarios is not thoroughly studied. In industries like smart retail, product information is often highly time-sensitive, as customers seek to purchase the latest styles. The application of large models needs to keep pace with current shopping trends and the current store inventories, posing a significant challenge. In contrast, conventional search engine technologies can track real-time product information, reviews, and trends. Therefore, in our research on intelligent services in unmanned retail, the approach is not merely to replicate existing functionalities of large models but to integrate them with a dynamic knowledge base.

For dynamic knowledge base, a similar work has been done by Cao et al.[23]. They studied how to build a knowledge base using automated methods. They proposed to extract knowledge points from textbooks. Their methods required a certain amount of data for pre-training in related topics. For our retail store applications, by updating the knowledge base with current product information, we aim to provide more real-time shopping services, addressing some of the limitations of current large model applications.

For personalized recommendation services, we need to comprehend users' shopping needs, analyze attributes and characteristics of product categories, match and recommend the most suitable products to users, and achieve guided services in smart retail. Many work has been done on the personalized recommendation. Liu et al.[24] proposed to use a novel keyword filtering algorithm to match children's book to children's fuzzy inquiries. In their work, the recommender system had high efficiency and the ability of understanding children's intention was promising. With the capability of large language models (LLMs) to comprehend user intentions, retail SKUs might be more effectively guided for the customer.

One of the major challenges with LLMs based retail application is the computational resource requirement. Especially for unmmaned retail industry, the cost needs to be limited to meet the market requirements. The computational efficiency was studied by previous works in retail object recognition[25–27]. They proposed to reduce the computational cost while maintain the recognition accuracy. It is important for reducing cost for unmanned retail store. One promising approach is fine-tuning (FT) large models with limited parameters. Aghajanyan et al.[28], studied intrinsic dimensionality in language models. In this work, they analyzed that a low-dimensional space is sufficient for incremental fine-tuning parameters for the majority of downstream jobs. This suggests that adapting to specific environments does not necessitate overly complex parameter updates.

To address the computational resource challenge, as well as the training data limitations, we proposed a new method for improving the fine-tuning efficiency for LLMs in smart retail scenarios. The space where model parameters reside exhibits a lower dimensionality. Even in overparameterized models, the model may effectively

vary within a smaller-dimensional subspace. This suggests that changes in model parameters can be represented by a lower-dimensional subspace. In particular, weight matrices can be approximated by lower-rank matrices. A low-rank representation implies that some rows or columns in the matrix can be expressed as linear combinations of other rows or columns. In this paper, we propose to further reduce the trainable parameters using metaheuristic approach, in order to improve the fine-tuning with limited domain specific training data.

Overall system design

The proposed end-to-end framework includes the following modules: fine-tuned language model, SKUs recognition model, SKUs database, customer database and recommender system, all working together to provide guidance to customers in an unmanned store. Here's a breakdown of the technical aspects of the system, as shown in Fig. 2.

Large language model

The pretrained language model is a type of artificial intelligence that has been trained on a large corpus of text data. It understands natural language and can perform various language-related tasks. In this system, the language model's role is to understand and process customer queries or requests related to finding specific products in the store.

Recognition of SKUs

Retail object recognition model is used to identify the products within the store environment. The YOLO-based retail object recognition module serves as a pivotal component in our framework, enabling efficient and real-time identification of SKUs. Leveraging the YOLO algorithm, our system ensures accurate and rapid detection of retail objects, enhancing the overall performance of the unmanned retail environment.

Language understanding and query processing

When a customer interacts with the system, the language model is responsible for understanding the intent and extracting relevant information. It processes the input and identifies keywords, product names, or queries related to finding products in the store.

SKUs recommender system

The system uses the extracted information from the customer's query to match it with the retail object features. It determines which products are relevant to the customer's request and provides recommendations based on this matching process.

The primary value and purpose of this system are to provide a convenient and efficient way for customers to navigate an unmanned store and find their desired products without limited human assistance. By leveraging AI technologies such as language understanding, object recognition, smart recommendation, the system promotes smart retail by: (i) allowing customers to interact with AI using natural language, (ii) assisting customers in locating products quickly and accurately, (iii) providing a personalized and helpful shopping suggestions, (iv) reducing the need for staff presence in the store, leading to cost savings and operational efficiency.

Methodology

To enhance the reliability of the large language models (LLMs), we incorporated a custom-built domain knowledge base. We constructed a comprehensive product information database that encompasses SKU titles, IDs, visual images, categories, product descriptions, purchase history, online reviews, and attribute tags. This knowledge

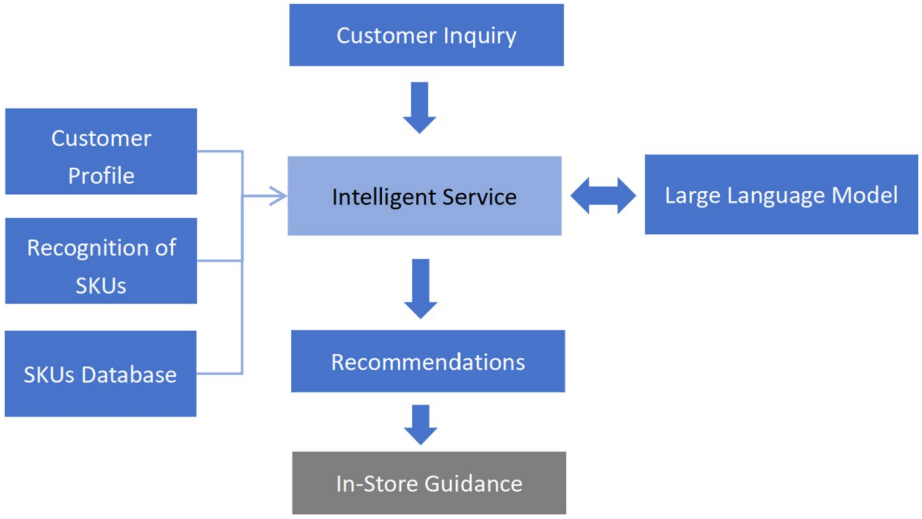


Fig. 2. System layout of the proposed service powered by large language model.

base serves as a valuable resource for the LLMs, providing a rich context and background information to improve its accuracy and effectiveness in understanding and responding to customer inquiries. The integration of this domain knowledge enhances the language model's ability to navigate and interpret information related to specific products, contributing to a more accurate and timely recommender system in unmanned retail stores.

The proposed framework encompasses the entire customer journey within unmanned intelligent stores, spanning from the initial interaction to locating the desired products. This service framework, extending from "customer to shelf," not only integrates advanced AI technologies but also establishes a dedicated domain-specific product knowledge base. This approach is not only significant for advancing the application of large models in the industry but also holds positive implications for driving innovation in the intelligent retail sector.

SKUs recognition model

In terms of product recognition, we study the You Only Look Once (YOLO) model, a residual neural network, for visual identification of SKUs (stock keeping units) on the store shelves. This approach utilizes the YOLO model to efficiently recognize and pinpoint the location of individual products within the inventory on the shelves. By employing a residual neural network architecture, we aim to enhance the accuracy and efficiency of the visual recognition process, providing a robust solution for SKU identification and localization in the context of unmanned retail environments.

Fine-tuning LLMs for retail intelligence

The philosophy behind our proposed method

The challenge in Fine-Tuning (FT) lies in specific downstream tasks where the training data is often limited, yet the model parameters are excessively large, violating the inherent constraints between them. Therefore, the proposed solution involves employing our algorithm to optimally reduce the trainable parameter size, thereby adhering to the empirical constraints imposed by the "model parameter size—training data size" relationship.

In our work, we constructed a shopping inquiry dataset tailored for unmanned retail scenarios, utilized for fine-tuning the LLM through supervised training to enhance its understanding of specific products within the store context. The retrained LLM was seamlessly integrated into the framework, establishing an end-to-end service model from the customer to the shelf. This approach aims to improve the language model's ability to interpret specific products within the context of customer inquiries, ultimately enhancing the overall intelligent service experience in unmanned retail settings.

Incremental fine-tuning can be categorized into three technical approaches: additive methods, prescriptive methods, and re-parameterization methods (Re-factor).

Additive methods

Continuing training on existing large models by introducing new domain-specific data gradually adapts the model to the specific requirements of the domain. This method helps retain previously acquired general knowledge while making the model more specialized.

Prescriptive methods

Adjusting the model's parameters or layers specifically for a particular domain to make it more suitable for handling specific types of information. This method focuses on fine-tuning the model's structure to enhance its performance on specific tasks.

Re-parameterization methods

Reconfiguring the model's parameters to align with the tasks in a specific domain. This may involve modifying the model's hierarchical structure or parameter configuration to optimize its performance.

Additionally, incremental FT possesses the advantageous feature of real-time hot swapping. This is particularly suitable for unmanned retail store applications, where different fine-tuned large models can be prepared for different stores or retail scenarios, allowing for immediate replacement and better meeting diverse specific requirements, thus improving the adaptability of the model.

Inspired by the Low-Rank Adaptation (LoRA) method[29], we propose a novel approach for fine-tuning large models. This method offers distinct advantages for limited training data set and limited computational resources. We propose to use metaheuristic algorithm to optimize the parameters in the lower dimensional space. The parameters are partially masked as fixed or trainable, as shown in Fig. 3, to accelerate the fine-tuning with less training dataset. In comparison to methods involving the addition of new parameters, our approach eliminates the need for inserting additional network modules. When compared to prescriptive methods, our fine-tuning performance excels, encompassing a broader range of fine-tuning parameters.

In LoRA, the low rank optimization is defined as follows[29]:

$$W_0 + \Delta W = W_0 + BA \tag{1}$$

where W is the weight, W_0 is the frozen weight parameters, $B \in \mathbb{R}^{d\times r}$, $A \in \mathbb{R}^{r\times k}$, d and k are the dimensions of original parameter matrix, r is the low rank of the decomposed space.

Specifically, through the optimization process of the low-dimensional parameter space (matrix A and B) in LoRA, a good initialization is crucial, especially given the limited amount of data. Empirically, it is assumed that fixing the parameters of the Mask is equivalent to providing a good approximation.

The encoding of parameters at mask positions is a direct binary sequence representation, bearing striking resemblance to the chromosome encoding in genetic algorithms (GA). In the low-rank space, parameter

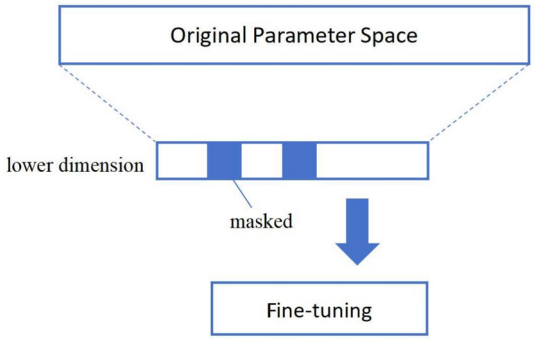


Fig. 3. Stochastic optimization in low dimensional parameter space.

representation is complete, albeit efficiently compressed. However, not every parameter necessitates fine-tuning, suggesting the existence of further engineering approximations.

Intuitions behind our optimization design

In the optimization of combined mask positions, during each fitness computation, we speculate the possibility of additional engineering approximations in the iterative steps. By identifying sensitive positions for the given batch of data early in the iteration process, we can selectively choose parameters at these sensitive positions for subsequent fine-tuning. This obviates the need for fine-tuning the entire set of parameters continuously.

In order to further optimize and efficiently retrain models while minimizing the demands on training data size, we employed stochastic optimization algorithms within the low-dimensional parameter space. Through this, we achieved a secondary compression, strategically selecting parameters at positions insensitive to training data to reduce the overall count of trainable parameters. Additionally, to economize the total computational load during the training process, we approximated the optimization cost function calculation. We constrained the iteration count of gradient descent during parameter updates, thereby advancing the computation of fitness values at an earlier stage.

Metaheuristic approach

We conducted a comparative validation of optimization effects using two algorithms, Genetic Algorithm (GA) and Shuffled Frog Leaping Algorithm (SFLA), as shown in Figs. 4 and 5.

Both in GA and SFLA, we take the following steps for the solution encoding for the population initialization. Parameter Vector C Initialization: Combine parameters A and B into a single vector C. This is done by concatenating the values of A and B.

$$C = [A_{flat}, B_{flat}] \tag{2}$$

where A_{flat}, B_{flat} are results from flatten operations.

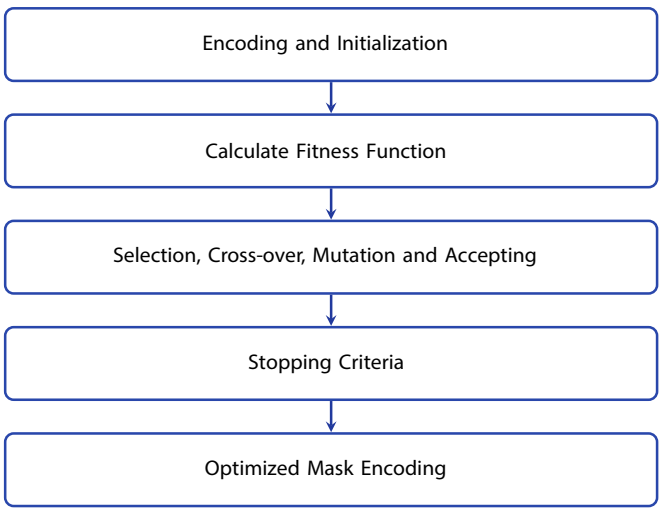


Fig. 4. Flow chart of the GA optimization.

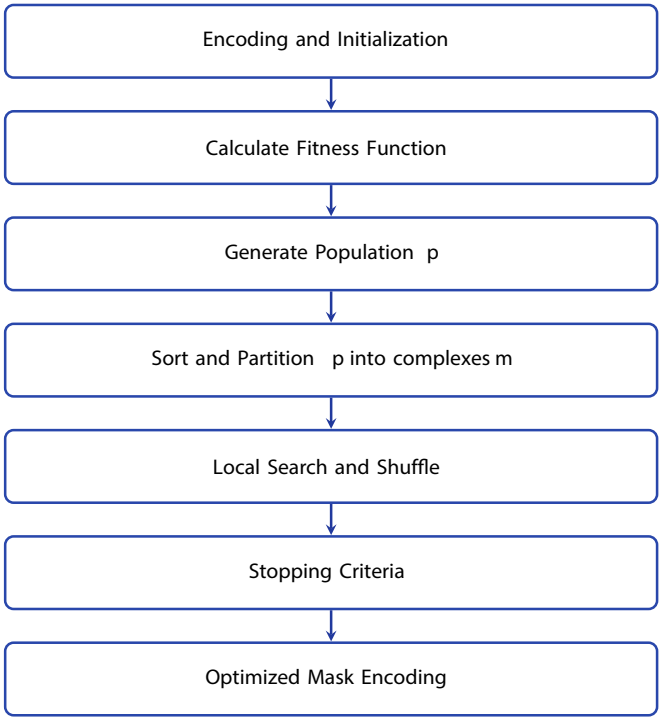


Fig. 5. Flow chart of the SFLA optimization.

Encoding mask sites into binary representation: assign the values in vector C to a binary representation M, where each bit corresponds to a mask site. 1 stands for masked, and 0 stands for not masked (trainable).

Let M_i be the ith element of M, then the binary representation M is given by:

$$M = \{M_i\}_{i=1}^{n} \tag{3}$$

Initialization of population: generate a population of potential solutions by creating individuals with different combinations of binary representation M.

The fitness function f(M)

Language model like BERT or GPT-2, can be formulated as the fitness function based on the model's performance on a set of training data in the recall task. The fitness value is defined from the loss metrics. Let's denote the model as $N(X, M)$, where X represents the input of the model. Let D be the training dataset, and Y be the true labels associated with the inputs in D. The fitness function $f(M)$ can be defined in terms of a loss function metric. Let's denote the loss as L for simplicity.

$$f(M) = -L(N(X, M), Y) \tag{4}$$

Optimization goal

The fitness function is formulated based on the performance of the model on the training data using a relevant loss function. It provides a quantitative measure of how well the model's parameters M fits the training data.

Different compression ratios for model training parameters were set, and the optimal selection of mask positions was determined through this approach. As shown in Fig. 6, we adopted SFLA which presented better results over GA. RoBERTa[30] and GPT-2[31] are used as the language models in our training. Experimental settings will be given in details in "Experimental results".

The SFLA algorithm, based on the metaheuristic mechanism, efficiently escapes local optima, enabling the identification of globally optimal combinations of mask positions. This approach circumvents the computational overhead associated with exhaustive search, facilitating the rapid discovery of optimal combinations. In each fitness computation iteration, multiple optimizations of fine-tuning parameters through gradient descent are required. Hence, an efficient combinatorial search algorithm is a crucial design element in achieving this.

Recommender module

In this module, we aim to utilize large language models and domain-specific knowledge for intelligent responses to customer shopping requirements in unmanned retail settings, particularly to compare with the conventional recommendation algorithms.

We use LLMs to understand the user inquiry and analyze the SKUs profile information, hence to match personalized needs in the recall stage of the recommender system. We leverage language models to establish semantic

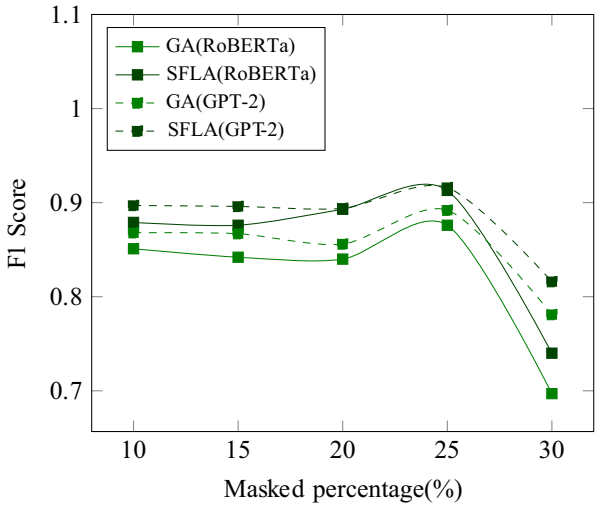


Fig. 6. Comparing GA and SFLA in M-LoRA fine-tuning: dashed line stands for GPT-2; darker color stands for SFLA.

connections between user query text and the semantic content within the product knowledge base. This involves associating information such as names, attributes, descriptions, and reviews from the product knowledge base with the semantics extracted from user inquiries. Through this process, we can effectively match and recall a list of relevant products that align with the user's query, enhancing the precision of our recommendation system.

The implementation of ranking in recommendations through the large language model method involves utilizing prompts to guide the language model's attention towards user profiles, item attributes, and reviews. In our prompt engineering instructions, we focus on factors including positive and negative reviews, user preference description, thereby enabling the proper ranking of items. This approach leverages the language model's ability to understand comments, contributing to an effective ranking that prioritizes user preferences and item characteristics. More experimental details are given in "Recommendations of SKUs".

The database

Visual database

We utilize two publicly available benchmark test datasets for our retail object recognition test: SKU-110K[7] and RPC[11].

The SKU-110K dataset comprises 11,762 images, featuring over 1.7 million annotated bounding boxes captured in dense scenes. This dataset is divided into 8233 training images, 588 images for validation, and 2941 images for the test set, totaling approximately 1,733,678 instances. The SKU-110K dataset features low-resolution original images with extensive variations, minimal differences between categories leading to similar shapes or colors among SKUs on the same shelf, and dense packing of products, with most images containing hundreds of objects. The images were sourced from numerous supermarket stores, exhibiting diverse proportions, viewing angles, lighting conditions, and noise levels. All images are adjusted to a megapixel resolution. In this dataset, most instances are closely packed, and object orientation falls within the range of $[-15°, 15°]$.

The RPC dataset comprises 200 retail product types and a total of 83,739 images, encompassing 53,739 single-product images and 30,000 multi-product images. These product types are classified into 17 categories, incorporating hierarchical structural information. In the single product setting, images are gathered with only one product positioned on a turntable. Conversely, checkout images are captured from a top view, featuring multiple products arranged together.

Recommendation database

To fine-tune our LLMs, user inputs (single or multiple instances) describing shopping needs and pertinent personalized information, including inquiries about specific products, are collected locally. Through semantic matching searches, product titles, descriptions, online reviews, and other textual information from the SKUs knowledge base are retrieved. A manual ranking of the top 5 recommended products is performed, with the best recommendation placed at the forefront. This constitutes the supervised information. A total of 11,800 instances, with 80% allocated for training, 10% for validation, and 10% for testing.

Statistic sample distributions, SKUs distribution and annotators distribution are shown in Figs. 7 and 8. The complete consistent annotations means that all annotators give the same result. Descriptive and Non-Descriptive mean whether the user inquiry contains details on what he or she wants to purchase. The annotator accuracy means the percentage of results that are accepted by a majority-voting rule.

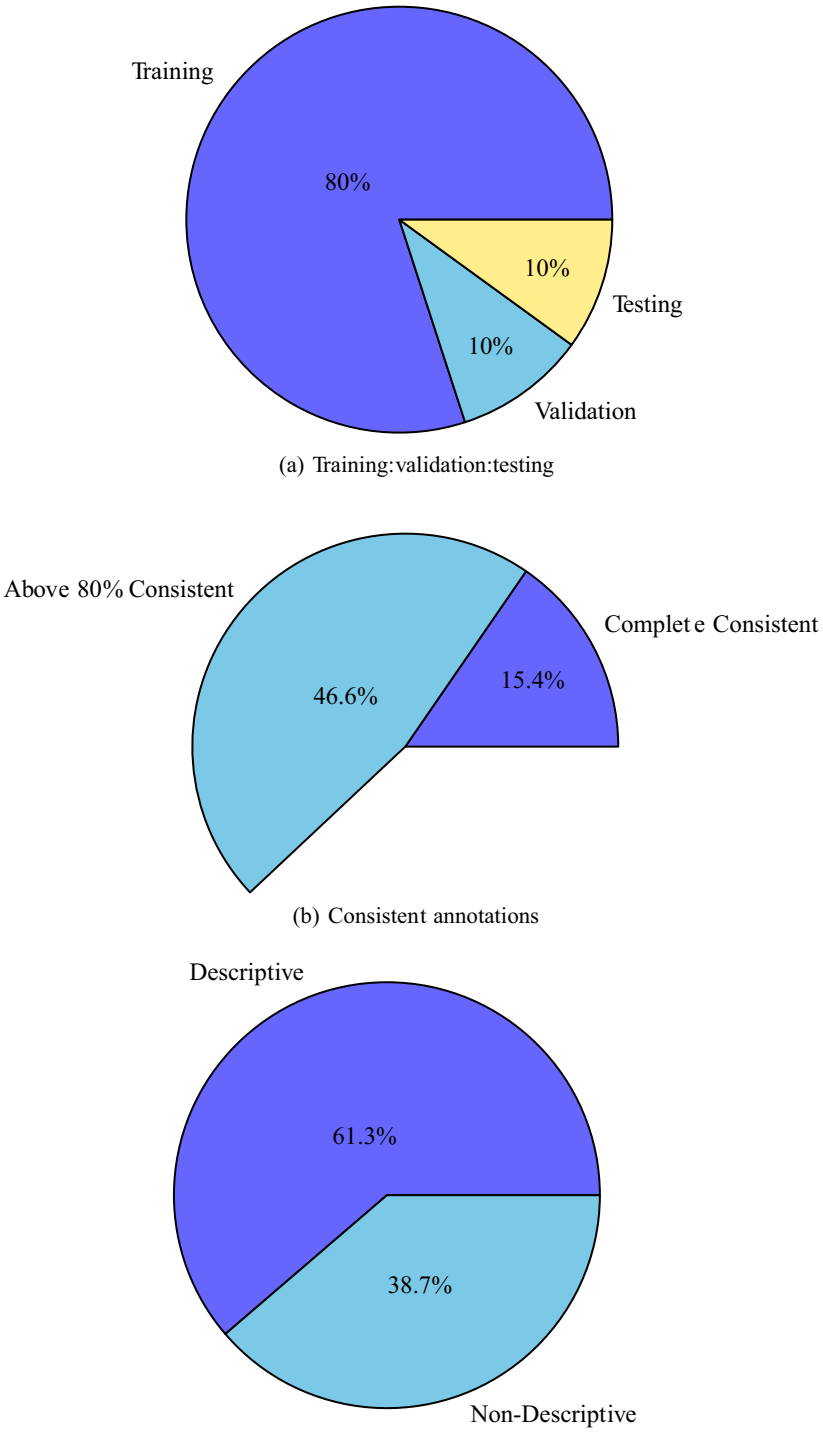
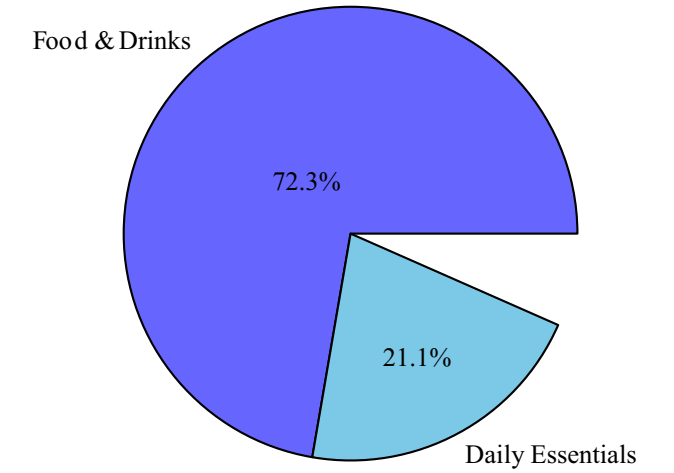


Fig. 7. Database details: sample distribution.

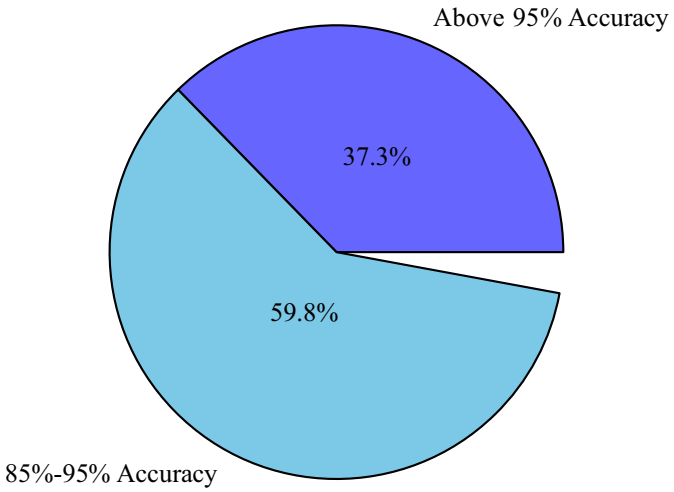
Experimental results

Visual recognition of SKUs

In this section, we conducted experiments comparing the performance of YOLO and several popular recognition methods in identifying retail product SKUs. The effectiveness of these methods in automatically recognizing retail products is a crucial step in unmanned retail services. Building on the validation of this step, we further combined large models by identifying semantic labels, enabling us to provide intelligent consultations and recommendations driven by these large models.



(a) SKU distribution



(b) Annotator distribution

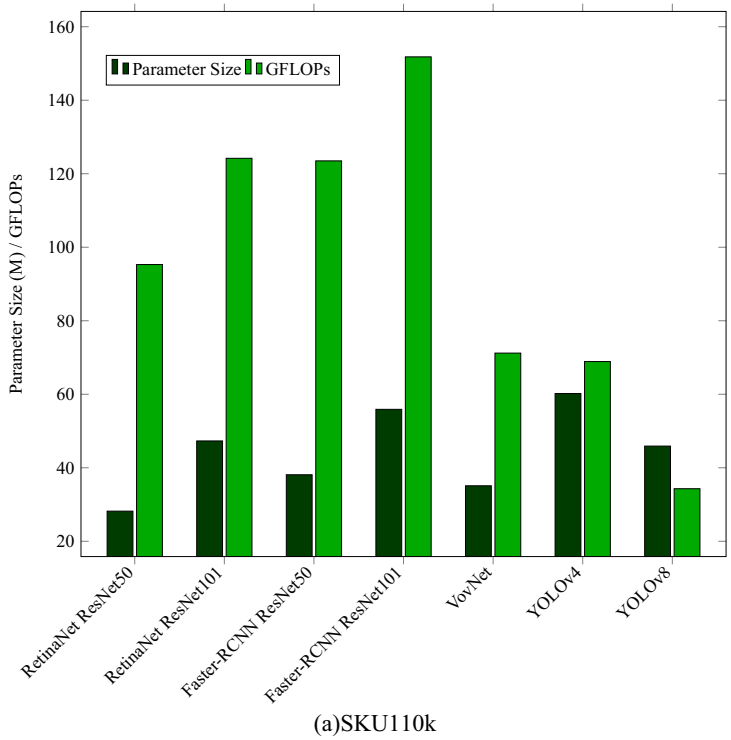
Fig. 8. Database details: SKUs & annotators.

Model architecture	Database	Top-1 accuracy (%)	Top-5 accuracy (%)	F1 score	mAP (%)	AP.75 (%)
RetinaNet-ResNet50	SKU110k	50.2	78.4	0.503	46.6	49.2
RetinaNet-ResNet101	SKU110k	57.7	85.6	0.515	48.3	55.1
FasterRCNN-ResNet50	SKU110k	56.1	83.4	0.516	48.1	54.9
FasterRCNN-ResNet101	SKU110k	55.8	83.0	0.531	49.2	53.1
VovNet	SKU110k	49.2	77.1	0.489	42.8	47.2
YOLOv4	SKU110k	64.5	88.9	0.557	51.1	62.1
YOLOv8	SKU110k	66.7	89.1	0.612	59.7	64.8
RetinaNet-ResNet50	RPC	76.1	88.3	0.741	72.5	75.5
RetinaNet-ResNet101	RPC	76.2	89.1	0.765	73.1	75.8
FasterRCNN-ResNet50	RPC	75.3	80.4	0.708	70.5	74.1
FasterRCNN-ResNet101	RPC	76.8	88.5	0.732	72.0	75.1
VovNet	RPC	74.8	80.1	0.701	69.7	73.3
YOLOv4	RPC	78.9	92.3	0.756	74.1	77.5
YOLOv8	RPC	80.1	94.2	0.789	76.2	79.3

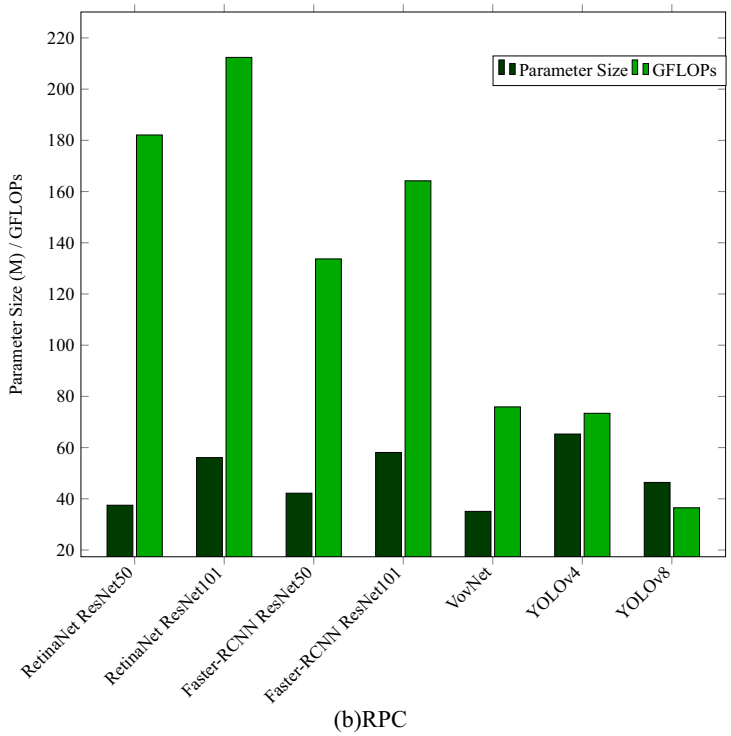
Table 1. Comparison of retail SKUs models on public databases.

As shown in Table 1, recognition metrics using various models are studied on two public databases, including RetinaNet[32], FasterRCNN[33], VovNet[34], YOLOv4[18] and YOLOv8[19]. The integration of visual models for automated SKU recognition lays the foundation for advanced applications, enhancing the capabilities of large models in delivering intelligent services within the unmanned retail sector.

Another challenge is the computational efficiency, as shown in Fig. 9, we tested the parameter size and GFLOPs(Giga Floating Point Operations Per Second)for each model. In unmanned retail applications, the process of shelving products, i.e., registering new arrivals, could involve updating the model with new categories



(a)SKU110k



(b)RPC

Fig. 9. SKUs recognition model efficiency comparison (SKU110k).

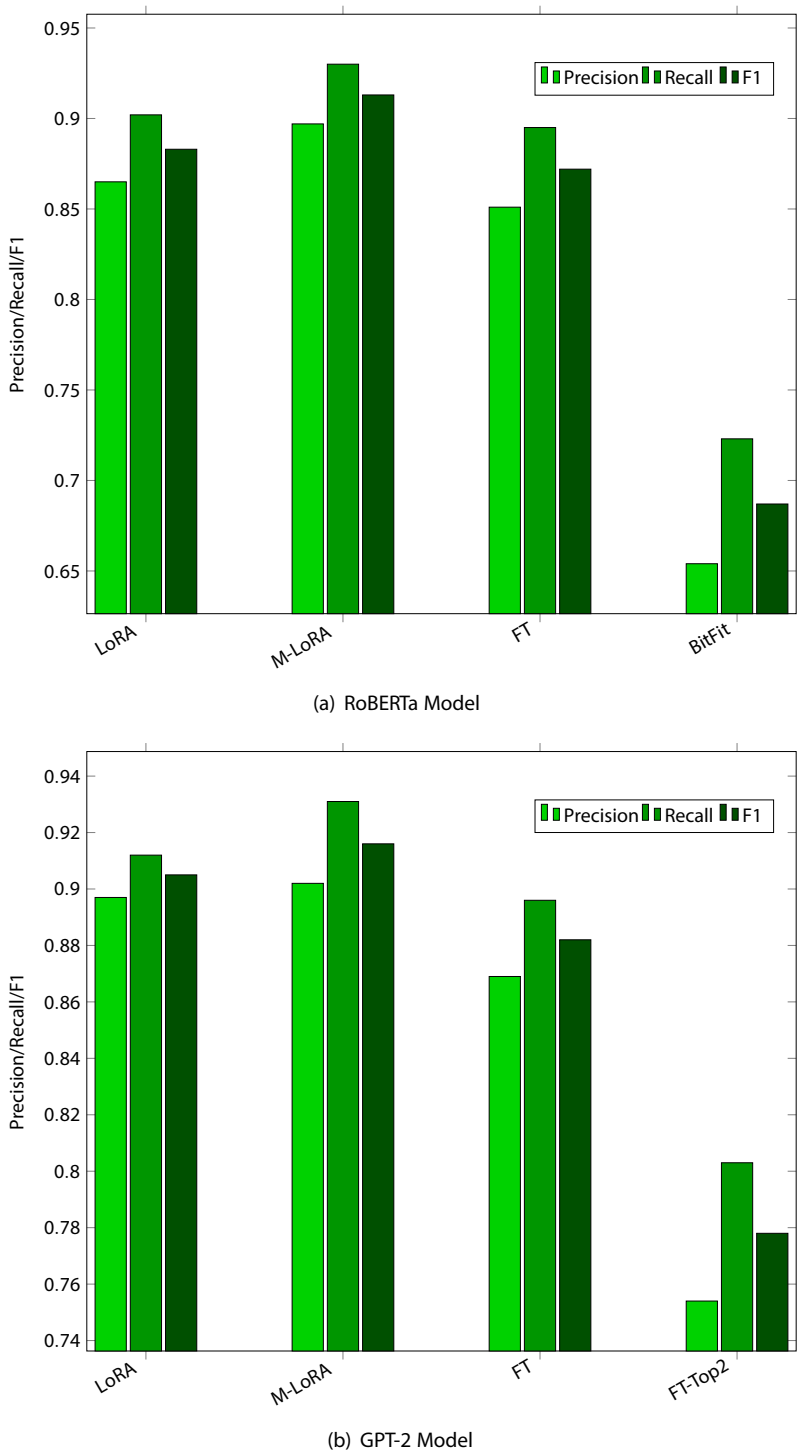


Fig. 10. Fine-tuning methods comparison: masked with 25%.

and conducting possible model re-training. Although we may choose product re-identification methods which requires no re-training cost, we still need to build new models when we deploy new retail stores to expand the business scale. In this phase, the efficient retraining of the model becomes crucial, considering the resources consumed during the process. Furthermore, in the context of product identification, the regular changes in shelf items due to sales activity necessitate continuous updates to the identification results in unmanned retail stores. Therefore, the computational speed during identification is also of paramount importance. Efficient methods for model retraining and fast computation during product identification are essential considerations, especially given the resource constraints and the dynamic nature of inventory in unmanned retail settings.

Method	Model	Parameter size (M)
LoRA	RoBERTa	0.3
M-LoRA	RoBERTa	0.21
FT	RoBERTa	125.0
BitFit	RoBERTa	0.1
LoRA	GPT-2	0.35
M-LoRA	GPT-2	0.245
FT	GPT-2	354.92
FT(Top2 layer)	GPT-2	25.19

Table 2. Model size using different fine-tuning methods (M-LoRA masked with 25%).

Method	Model	Mask percentage (%)	Parameter size	Precision	Recall	F1
LoRA	RoBERTa	–	0.3M	0.865	0.902	0.883
M-LoRA	RoBERTa	10	0.27M	0.865	0.889	0.879
M-LoRA	RoBERTa	15	0.26M	0.864	0.889	0.876
M-LoRA	RoBERTa	20	0.24M	0.876	0.911	0.893
M-LoRA	RoBERTa	25	0.23M	0.897	0.930	0.913
M-LoRA	RoBERTa	30	0.21M	0.701	0.784	0.740
LoRA	GPT-2	–	0.35M	0.897	0.912	0.905
M-LoRA	GPT-2	10	0.32M	0.886	0.908	0.897
M-LoRA	GPT-2	15	0.30M	0.886	0.907	0.896
M-LoRA	GPT-2	20	0.28M	0.885	0.904	0.894
M-LoRA	GPT-2	25	0.26M	0.902	0.931	0.916
M-LoRA	GPT-2	30	0.25M	0.783	0.851	0.816

Table 3. Summary of recall tests under different mask percentages of M-LoRA.

User type	SKUs type	Recall	Precision	F1
Descriptive	Food & drinks	0.912	0.937	0.923
Non-descriptive	Food & drinks	0.881	0.915	0.898
Descriptive	Daily essentials	0.869	0.901	0.885
Non-descriptive	Daily essentials	0.861	0.903	0.881
Descriptive	Other groceries	0.910	0.935	0.922
Non-descriptive	Other groceries	0.891	0.909	0.900

Table 4. LLM-based recommendation: recall results.

Fine-tuning large models for smart recommendation

Through the random projection algorithm presented in this study, we further reduced the demand for parameters, achieving a reduction of over 25% in parameter quantity. Importantly, this reduction was achieved while maintaining a comparable level of recognition performance.

In our experiment, we update base model of the BERT family, and GPT-2. We primarily consider the resource constraints in retail store industry, the construction cost of unmanned retail stores is a crucial consideration. Consequently, we opted not to invest in expensive hardware and focused on smaller, efficient models. The experiment is carried out on cloud-based GPUs. We use NVIDIA Tesla V100 for modeling experiments.

Our fine-tuning method demonstrates greater efficiency. A comparison between LoRA and our masked-LoRA (M-LoRA) is shown in Fig. 10 employing our proposed method alongside various other mainstream methods allows for a comparison of the improvement in F1 metric scores.

RoBERTa (robustly optimized BERT approach), base model, is an enhanced and optimized version built upon the BERT (Bidirectional Encoder Representations from Transformers) model.

GPT-2 (Generative Pre-trained Transformer 2), is a state-of-the-art language model developed by OpenAI. It is part of the GPT series, known for its ability to generate coherent and contextually relevant text. Due to the limited retail store computational resources, we choose the medium size version of GPT-2 for smart SKUs recommendation.

Trainable parameters are much smaller than the full set of parameters in the language model, hence to improve the model training efficiency. As shown in Table 2, the proposed M-LoRA is compared with the original LoRA, FT (full fine-tuning), FT (Top2 layers) and BitFit[35] methods. BitFit method simply trains the bias vectors while freezing other parameters.

In order to better understanding the effect of the randomized masked parameters, we observed the change in F1 score by adjusting the extent of the mask, as shown in Table 3.

Recommendations of SKUs

In this experiment, building upon the validated visual recognition of SKU items, we enhanced the application effectiveness of large models in the specific domain of unmanned retail by incorporating a local product knowledge base based on the semantic labels of products. We validated the intelligent response to customer shopping needs using large models and the knowledge base, providing personalized recommendations for products on unmanned shelves within the store. Additionally, we compared this approach with traditional recommendation algorithms.

In the recommender system verification, we compared the proposed system with conventional models. For the data test, we have human annotators who act as the store attendants to annotate SKUs selections. Their task involved reviewing customer inquiry texts and selecting the appropriate recommended products. The human annotations serve as the ground-truth.

For other existing recommender approaches based on operational user behavior data, we need to expand user shopping behavior, in the future operations. This will require the accumulation of a substantial amount of unmanned store operational data before implementation. Currently, we are using the data generated by human annotators.

Recall experiment

The goal of the recall experiment is to evaluate the performance of the recommendation system during the recall phase. The testing dataset includes user inquiry records, retail product titles, categories, attributes tags, and description information. The dataset is divided into training, validation, and test sets (as described in the previous section on LLMs experimental conditions).

The LLM-driven approach is employed, and different traditional recommendation models are compared. Recall experiments are conducted for each trained model, returning the top K recommendations for each user, set to 5 in this case. Precision, Recall, and F1 metrics are used to evaluate the recall performance, as shown in Table 4.

Ranking experiment

The ranking experiment aims to further optimize our recommendation effectiveness, ensuring that the recommended products align more closely with users' shopping expectations. The goal is to have the most interesting products for the users appear at the forefront of the recommended search results, making it more applicable to real-world scenarios.

In terms of model comparison, various classic recommendation system ranking models are considered, including logistic regression (LR), deep neural network (DNN), wide & deep[36], and DeepFM[37], as shown in Fig.11.

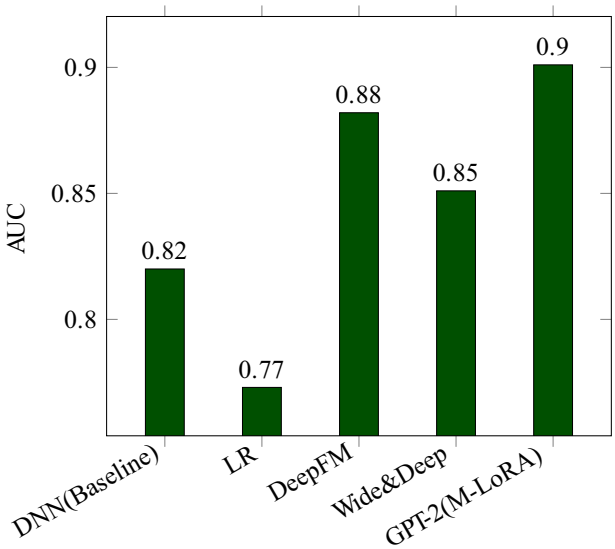


Fig. 11. SKUs recommendation ranking results.

Conclusions

Unmanned intelligent stores are increasingly gaining attention due to their high efficiency and convenience. However, achieving a seamless intelligent customer service experience in unmanned retail scenarios involves addressing two key issues: understanding customer needs and guiding them to the appropriate shelf locations. This study aims to apply large language models to intelligent analysis of customer demands and employ computer vision technology for product recognition and positioning. Our main contribution is to construct an "end-to-end" (customer to shelf) unmanned retail intelligent service system, and proposed an efficient algorithm for data-limited downstream job training in masked low rank parameter space.

Data availability

The datasets used in this study are available in the public databases: RPC dataset, https://rpc-dataset.github.io/, SKU-110K dataset, https://www.kaggle.com/datasets/thedatasith/sku110k-annotations

Received: 23 January 2024; Accepted: 23 August 2024
Published online: 27 August 2024

References

1. Piovani, D., Zachariadis, V. & Batty, M. Quantifying retail agglomeration using diverse spatial data. *Sci. Rep.* **7**, 5451 (2017).
2. Hofman, O. et al. X-detect: Explainable adversarial patch detection for object detectors in retail. arXiv preprint arXiv:2306.08422 (2023).
3. Dhonde, A., Guntur, P. & Palani, V. Adaptive ROI with pretrained models for automated retail checkout. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5506–5509 (2023).
4. De Biasio, A. *Retail Shelf Analytics Through Image Processing and Deep Learning* (Universityo Padua, 2019).
5. Tonioni, A. & Di Stefano, L. Domain invariant hierarchical embedding for grocery products recognition. *Comput. Vis. Image Underst.* **182**, 81–92 (2019).
6. Wei, Y. et al. Deep learning for retail product recognition: Challenges and techniques. *Comput. Intell. Neurosci.* **2020** (2020).
7. Goldman, E., Herzig, R., Eisenschtat, A., Goldberger, J. & Hassner, T. Precise detection in densely packed scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5227–5236 (2019).
8. Geng, W. et al. Fine-grained grocery product recognition by one-shot learning. In *Proceedings of the 26th ACM International Conference on Multimedia*. 1706–1714 (2018).
9. Wang, C., Huang, C., Zhu, X. & Zhao, L. One-shot retail product identification based on improved siamese neural networks. *Circuits Syst. Signal Process.* **41**, 6098–6112 (2022).
10. Wang, C., Huang, C., Zhu, X., Li, Z. & Zhao, L. Smart retail skus checkout using improved residual network. *Sci. Rep.* **13**, 22512 (2023).
11. Wei, X.-S., Cui, Q., Yang, L., Wang, P. & Liu, L. Rpc: A large-scale retail product checkout dataset. arXiv preprint arXiv:1901.07249 (2019).
12. Santra, B., Shaw, A. K. & Mukherjee, D. P. Part-based annotation-free fine-grained classification of images of retail products. *Pattern Recognit.* **121**, 108257 (2022).
13. Dworakowski, D., Thompson, C., Pham-Hung, M. & Nejat, G. A robot architecture using contextslam to find products in unknown crowded retail environments. *Robotics* **10**, 110 (2021).
14. Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M. & Poggio, T. Robust object recognition with cortex-like mechanisms. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**, 411–426 (2007).
15. Huang, C. & Jiang, H. Image indexing and content analysis in childrenâ€™s picture books using a large-scale database. *Multimed. Tools Appl.* **78**, 20679–20695 (2019).
16. Huang, C. et al. Facial landmark configuration for improved detection. In *2012 IEEE International Workshop on Information Forensics and Security* (*WIFS*).
17. He, K., Zhang, X., Ren, S. & Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 770–778 (2016).
18. Bochkovskiy, A., Wang, C. & Liao, H. Y. M. Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934 (2020).
19. Talaat, F. & ZainEldin, H. An improved fire detection approach based on yolo-v8 for smart cities. *Neural Comput. Appl.* **35**(20939–20954), 7 (2023).
20. Bochkovskiy, A., Wang, C.-Y. & Liao, H.-Y. M. Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934 (2020).
21. Yin, Y., Li, H. & Fu, W. Faster-yolo: An accurate and faster object detection method. *Digital Signal Process.* **102**, 102756 (2020).
22. Leo, M., Carcagni, P. & Distante, C. A systematic investigation on end-to-end deep recognition of grocery products in the wild. In *2020 25th International Conference on Pattern Recognition* (*ICPR*). 7234–7241 (IEEE Computer Society, 2021).
23. Cao, L., Huang, C., Wang, Q. & Zhu, X. Knowledge point entity and relation extraction based on pre-training model in education resources. Preprints (2023).
24. Liu, Y., Gao, T., Song, B. & Huang, C. Personalized recommender system for children's book recommendation with a real time interactive robot. *J. Data Sci. Intell. Syst.* (2023).
25. Wang, J., Huang, C., Zhao, L. & Li, Z. Lightweight identification of retail products based on improved convolutional neural network. *Multimed. Tools Appl.* **81**, 31313–31328 (2022).
26. Horng, S.-J. & Huang, P.-S. Building unmanned store identification systems using yolov4 and siamese network. *Appl. Sci.* **12**, 3826 (2022).
27. Cao, Y., Ikenoya, Y., Kawaguchi, T., Hashimoto, S. & Morino, T. A real-time application for the analysis of multi-purpose vending machines with machine learning. *Sensors* **23**, 1935 (2023).
28. Aghajanyan, A., Gupta, S. & Zettlemoyer, L. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*. Vol. 1: Long Papers. 7319–7328 (2021).
29. Hu, E. J. et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations* (2021).
30. Liu, Y. et al. Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 (2019).
31. Radford, A. *et al.* Language models are unsupervised multitask learners. *OpenAI Blog* **1**, 9 (2019).
32. Lin, T.-Y., Goyal, P., Girshick, R., He, K. & Dollár, P. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision*. 2980–2988 (IEEE, 2017).
33. Ren, S., He, K., Girshick, R. & Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **28**, 91–99 (2015).

34. Lee, Y., Hwang, J., Lee, S., Bae, Y. & Park, J. An energy and GPU-computation efficient backbone network for real-time object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops* (2019).
35. Zaken, E. B., Ravfogel, S. & Goldberg, Y. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. arXiv preprint arXiv:2106.10199 (2021).
36. Cheng, H. et al. Wide & deep learning for recommender systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*. 7–10 (2016).
37. Guo, H., Tang, R., Ye, Y., Li, Z. & He, X. Deepfm: A factorization-machine based neural network for ctr prediction. arXiv preprint arXiv:1703.04247 (2017).

Author contributions
P.Z. and W.W. conceived the methods, W.W., P.Z. and C.S. conducted the experiments, W.W., P.Z., C.S. and D.F. analysed the results. All authors reviewed the manuscript.

Competing interests
The authors declare no competing interests.

Additional information
Correspondence and requests for materials should be addressed to P.Z.

Reprints and permissions information is available at www.nature.com/reprints.

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