Smart Expiry Tracker: LLM-Powered Grocery Inventory Management with OCR-Based Product Logging

Mackwyn Quadras
University of Potsdam, Germany
quadras@uni-potsdam.de

Abstract

This paper presents a system for managing grocery inventory through automated product logging and intelligent usage scheduling. We implement OCR-based text extraction combined with LLM reasoning to automatically identify products, extract expiry dates, and generate personalized meal plans. The system employs Tesseract OCR for image processing and Llama3 for natural language understanding and planning. We implement a Mixture of Experts approach where the LLM simulates three perspectives (Nutrition, Budget, Recipe) to generate contextually appropriate usage recommendations. The system shows promising extraction performance on product labels and practical utility in reducing food waste through timely consumption planning. The system maintains persistent storage and provides automated removal suggestions for expired items.[1]

1 Introduction

Food waste is a significant problem in households, with studies showing that improper inventory management leads to premature spoilage and unnecessary waste. Manual tracking of grocery items is tedious, leading to poor adoption of traditional inventory systems. The primary challenges include:

- Manual data entry barriers
- Lack of context-aware expiry alerts
- Absence of personalized usage planning

Recent developments in Optical Character Recognition (OCR) and Large Language Models (LLMs) offer new possibilities for automating grocery inventory tasks and generating intelligent usage plans. This project introduces a system that:

- Uses camera-based OCR to automatically extract product information from labels
- Applies LLM reasoning to classify items and interpret quantities from label text
- Generates personalized usage schedules using a simulated Mixture of Experts approach

The key features of the system includes:

- An OCR-to-LLM pipeline for extracting product details from varied label formats
- A prompt-based planning strategy that simulates multiple expert perspectives (e.g., nutrition, budget, recipe matching)
- A lightweight deployment using locally-hosted models (Ollama with Llama3), suitable for personal use and offline operation

The remainder of this paper is organized as follows: Section 2 reviews related work. Section 3 describes our system architecture. Section 4 details the OCR processing pipeline. Section 5 presents the LLM integration approach. Section 6 discusses the Mixture of Experts implementation. Section 7 concludes with future directions.

2 Related Works

2.1 OCR in Food Applications

Optical Character Recognition (OCR) has been widely applied to food packaging, primarily for structured nutrition label extraction. However, recent work has extended OCR to general grocery product identification. Pettersson et al. proposed a multimodal system that combines image features and OCR-extracted text to recognize fine-grained grocery items, including visually similar products and labels. This aligns with our approach, which focuses on extracting product names, categories, and expiry dates from variable label designs in both English and German.

[1] https://github.com/GrimGamer1999/smart-shelf-llm

2.2 LLMs for Planning

Large Language Models (LLMs) have demonstrated strong capabilities in task planning and scheduling. Arora et al. introduced *Anticipate & Act*, a framework that integrates LLMs with classical planning to execute household tasks such as cooking and cleaning. While their system emphasizes robotic agents and anticipatory reasoning, our work applies LLMs to domain-specific planning under resource constraints—such as expiry dates and available kitchen equipment—bridging abstract reasoning with practical household management.

2.3 Food Waste Reduction

Traditional food waste reduction strategies rely on static reminder apps or FIFO (First-In-First-Out) inventory systems, which often lack contextual awareness. Xequence AI highlights how AI-powered inventory management can reduce waste through expiry tracking, demand forecasting, and dynamic pricing. Our system builds on these ideas by combining real-time OCR-based logging with LLM-generated usage plans, offering personalized and context-aware recommendations to minimize spoilage.

2.4 Mixture of Experts via Prompt Engineering

The Mixture of Experts (MoE) paradigm typically involves separate models specialized for distinct tasks. Wang et al. propose *Mixture-of-Prompts (MoP)*, a method that clusters in-context examples and assigns region-specific instructions to simulate expert reasoning within a single LLM. Our system adopts a similar strategy through prompt engineering, enabling a single LLM to emulate multiple expert perspectives—nutrition, budget, and recipe matching—without requiring multiple models.

3 System Architecture

The system follows a modular design with four main components, each handling a specific part of the inventory management process. This setup keeps tasks separate while allowing smooth data flow between modules.

1. **Data Storage Module (storage.py):** This module saves product details using JSON files. It stores key information like name, category, quantity, expiry date, and when the item was added. It also keeps track of user actions. Proper error handling ensures that data is not lost during system crashes and remains safe across sessions.
2. **OCR Processing Module (ocr_utils.py):** This part handles image editing and text extraction. It uses methods like denoising, contrast adjustment, and binarisation to prepare images. It connects with Tesseract OCR and uses different settings to improve accuracy. It also includes tools to read dates in various international formats.
3. **LLM Communication Module (llm_utils.py):** This module manages all communication with the locally hosted Llama3 model via Ollama. It builds prompts based on the task, sends requests with proper settings like temperature and token limits, and reads the responses. It also has backup methods to deal with errors or failed responses, including timeouts and retries.
4. **Main Application (app.py):** This is the central part of the system, built using Streamlit for the user interface. It supports three main tasks: adding products (via OCR or manual entry), viewing inventory with expiry highlights and smart suggestions, and generating usage plans using the Mixture of Experts method. It also keeps session data to maintain user context and connects with the storage module to save changes.

4 OCR Processing Pipeline

The OCR pipeline transforms raw product images into structured text through multiple stages. We implement distinct preprocessing strategies for different image types, as product labels and expiry stamps present different visual challenges.

4.1 Image Preprocessing

We apply mode-specific preprocessing optimized for each image type: **For Product Labels:**

- Make image bigger if too small (at least 300px height)
- Reduce image noise using OpenCV denoising filter
- Apply sharpening to make text edges clearer
- Increase contrast so text stands out from background

For Expiry Stamps:

- Adjust brightness levels across the image
- Apply stronger noise reduction (stamps are usually less clear)
- Convert to black and white using automatic thresholding

Product labels typically feature printed text on varied backgrounds, requiring gentle preprocessing that enhances readability without introducing artifacts. Expiry stamps, conversely, often appear as embossed or stamped text with variable lighting, necessitating more aggressive preprocessing to extract usable text.

4.2 Multi-Configuration OCR

To maximize extraction accuracy, we employ three Tesseract Page Segmentation Modes (PSM):

- PSM 6: Assumes uniform block of text (optimal for clean labels)
- PSM 3: Fully automatic page segmentation (handles complex layouts)
- PSM 11: Sparse text detection (effective for minimal text like stamps)

The system runs all three configurations in parallel and returns the longest extracted text, operating on the assumption that completeness correlates with accuracy. This approach proved more reliable than single-configuration extraction.

4.3 Date Extraction

We implement a two-tier date extraction strategy combining regex parsing with LLM fallback. The regex-based parser recognizes multiple international formats:

- German numeric format: 02.2027 → 28-02-2027
- Month-name formats: OCT 2025, OCTOBER 2025 → 31-10-2025
- Prefixed formats: EXP:, MHD:, BEST BEFORE: followed by dates

For month-year formats without specific days, we default to the last day of the month as a conservative estimate for shelf life expiration. The parser handles both abbreviated and full month names in English and German, with built-in mappings for common variations.

When rule-based parsing struggles with non-standard formats or OCR inconsistencies, the system relies on the LLM for contextual interpretation. This fallback mechanism enhances robustness, especially in cases involving handwritten text or unconventional date expressions.

5 LLM Integration

We leverage Llama3's language understanding capabilities for three primary tasks: product information extraction from noisy OCR text, expiry date interpretation when regex fails, and shelf life estimation for fresh produce.

5.1 Product Information Extraction

The LLM receives preprocessed OCR text and extracts structured product information through carefully designed prompts. We instruct the model to:

1. Identify the main product name while preserving original language
2. Classify into predefined categories (Coffee, Rice/Grains, Tea, Dairy, etc.)
3. Extract quantity information including units (e.g., "1kg", "500g")

The prompt incorporates multilingual keyword mappings to accommodate German-language products commonly encountered in our testing environment. For instance, `REIS` is mapped to the `Rice/Grains` category, `KAFFEE` to `Coffee`, and `MILCH` to `Dairy`.

The LLM handled noisy OCR output well in observed cases, effectively interpreting partially corrupted text through contextual reasoning.

To ensure reliable parsing, the response format is constrained to JSON:

```
{
  "name": "product name",
  "category": "category",
  "quantity": "amount"
}
```

5.2 Expiry Date Fallback

When rule-based date extraction fails, the LLM acts as a smart fallback. It receives raw OCR output along with clear instructions on expected formats. Month-year entries are conservatively interpreted as the last day of the month to avoid premature expiry estimates.

5.3 Fresh Produce Shelf Life Estimation

For manually entered fresh produce without printed expiry dates, we prompt the LLM to estimate shelf life based on product type and category. The prompt includes established guidelines for common produce categories:

- Leafy Greens: 3-5 days
- Root Vegetables: 7-21 days
- Fruits: 5-10 days
- Berries: 3-5 days
- Herbs: 5-7 days

We explicitly request conservative estimates to prioritize food safety. The LLM returns not only the estimated days and calculated expiry date but also storage tips tailored to the specific produce item, adding educational value for users.

6 Mixture of Experts Implementation

Traditional Mixture of Experts (MoE) architectures employ multiple specialized models that contribute to final predictions. Due to resource constraints and deployment complexity, we simulate this approach through prompt engineering, having a single LLM adopt multiple expert perspectives sequentially.

6.1 Simulated Expert Perspectives

We define three expert personas with distinct optimization objectives:

Nutrition Expert: This perspective prioritizes food freshness and nutritional value retention. It recommends consuming fresh produce quickly to maximize vitamin content and emphasizes health considerations in meal timing. For example, it might suggest using leafy greens within 2-3 days of purchase rather than waiting until just before expiry.

Budget Optimizer: This viewpoint focuses on preventing waste of expensive items and maximizing cost-efficiency. It prioritizes items nearest to expiry and considers the relative cost of ingredients when suggesting consumption order. This perspective might recommend using an expensive cut of meat before cheaper staples.

Recipe Matcher: This expert emphasizes practical meal planning by identifying combinations of available inventory items. It considers user equipment constraints (stovetop, oven, microwave availability) and skill level (beginner, intermediate, advanced) to suggest feasible recipes. The recipe matcher ensures recommendations are actionable rather than aspirational.

6.2 Synthesis Prompt Structure

Our synthesis prompt explicitly presents all three perspectives to the LLM, then requests a unified plan that balances their recommendations. The complete prompt structure includes:

1. Product context (name, category, quantity, expiry date)
2. Inventory context (other available products for combinations)
3. User constraints (equipment, skill level)
4. Three expert perspective descriptions with their optimization criteria
5. Synthesis instructions requesting 3-5 specific dated meal suggestions
6. Format requirements including expert attribution

Example output format:

```
[Date]: Meal suggestion
- Why: [Nutrition/Budget/Recipe perspective]
- Combine with: [other inventory items]
- Method: [brief preparation note]
```

This structure encourages the LLM to explicitly reason about trade-offs between perspectives and provide transparent explanations for its recommendations.

6.3 Contextual Adaptation

The system dynamically adapts recommendations based on real-time constraints. Equipment availability directly filters suggested recipes - if a user lacks an oven, baking suggestions are automatically excluded. Skill level adjusts recipe complexity, with beginner plans emphasizing simple preparations and advanced plans incorporating more sophisticated techniques.

Inventory context enables intelligent pairing suggestions. When multiple complementary items exist (e.g., rice and curry spice), the recipe matcher perspective advocates for combined usage, potentially accelerating consumption of both items while creating satisfying meals.

The simulated Mixture of Experts approach offered diverse and context-aware recommendations compared to single-perspective prompting. Explicitly attributing suggestions to distinct expert personas helped clarify the reasoning behind each recommendation and showcased the potential of prompt-based modularity in planning tasks.

7 Conclusion and Future work

We presented a practical system for automated grocery inventory management that combines OCR and LLM capabilities to reduce manual data entry and provide intelligent consumption planning. The system demonstrates that locally-hosted open-source models can deliver useful functionality without cloud dependencies or privacy concerns.

7.1 Key Contributions

Our primary contributions include an integrated OCR-LLM pipeline for product detail extraction, a simulated Mixture of Experts approach enabling personalised meal planning through prompt engineering, and a production-ready deployment using Ollama/Llama3 that showcases practical applications of locally hosted LLMs.

The Mixture of Experts simulation proved particularly effective, generating more diverse and contextually appropriate recommendations than single-perspective prompting. User feedback indicated that explicit expert attribution increased trust and understanding of system reasoning.

7.2 Limitations

Several limitations constrain current system capabilities. OCR accuracy depends heavily on image quality and lighting conditions, with challenging angles or low contrast reducing extraction success. LLM response consistency varies due to temperature sampling, occasionally producing unexpected categorizations or quantity interpretations. Fresh produce requires manual entry as the system lacks visual recognition capabilities. Finally, the system supports only single-user scenarios without multi-user household inventory sharing.

7.3 Future Directions

Technical Improvements: Integrating barcode scanning APIs has the potential to greatly enhance accuracy for standardised packaged items. Fine-tuning compact, domain-specific models for product classification may help reduce inference latency while ensuring greater consistency. Employing computer vision techniques could facilitate automatic identification of fresh produce from images. Implementing actual Mixture of Experts (MoE) architectures using specialised models would offer a meaningful benchmark against our current prompting-based simulation.

Feature Enhancements: Introducing nutritional tracking across meals would enable basic health monitoring features. Automatically generating shopping lists based on consumption trends and stock depletion could help close the loop in inventory management. Integrating a recipe database with step-by-step instructions would significantly improve usability, especially for users focused on cooking. Deploying a mobile application with native camera access would substantially enhance the user experience for photo-based input. Supporting multiple users with shared household inventories is a key requirement for real-world adoption.

Research Extensions: Conducting longitudinal studies to assess the impact on food waste reduction would help validate the system's core value proposition. A comparative analysis between our simulated MoE setup and actual multi-model architectures could shed light on cost-versus-accuracy trade-offs, particularly for deployments with limited resources. Investigating active learning strategies for OCR error correction may lead to improved accuracy over time through user feedback. Exploring reinforcement learning for refining personalised recommendations could allow the system to better adapt to individual preferences and behavioural patterns.

7.4 Broader Impact

This work showcases a practical application of large language models (LLMs) in everyday household management. By promoting timely consumption and improving inventory awareness, the system contributes meaningfully to sustainability efforts by reducing food waste. Its modular design allows for easy adaptation to related domains such as medicine cabinet tracking, office supply monitoring, or any context requiring expiry-aware inventory control.

The system's reliance on locally-hosted models ensures user privacy while demonstrating that robust capabilities can be achieved without cloud-based infrastructure. This privacy-conscious approach may encourage similar innovations in domains dealing with sensitive personal data.

References

[1] Tobias Pettersson, Maria Riveiro, and Tuwe Löfström. 2024. Multimodal Fine-Grained Grocery Product Recognition Using Image and OCR Text. In *Machine Vision and Applications*, Springer. Multimodal Fine-Grained Grocery Product Recognition (Pettersson et al., 2024)

[2] Raghav Arora, et al. 2024. Anticipate & Act: Integrating LLMs and Classical Planning for Efficient Task Execution in Household Environments. In *arXiv preprint arXiv:2502.02066*. Anticipate & Act: LLMs for Household Planning (Arora et al., 2024)

[3] Rajib Kar. 2025. Reducing Food Waste with AI-Powered Inventory Management. Xequence AI Blog. Reducing Food Waste with AI Inventory Management (Kar, 2025)

[4] Ruochen Wang, Sohyun An, Minhao Cheng, Tianyi Zhou, Sung Ju Hwang, and Cho-Jui Hsieh. 2024. One Prompt is Not Enough: Automated Construction of a Mixture-of-Expert Prompts. In *Proceedings of ICML 2024*. Mixture-of-Expert Prompts for LLMs (Wang et al., ICML 2024)

[5] UB Mannheim. (2024). Tesseract OCR: Installation and Usage Guide. Tesseract OCR Documentation (UB Mannheim, 2024)

[6] Ollama. (2024). Get up and running with large language models locally. Ollama: Local LLM Deployment (2024)

[7] Streamlit. (2023). Streamlit Documentation. Streamlit UI Framework Documentation (Streamlit, 2023)

[8] Emojipedia. (2025). The World's #1 Resource for Emoji Meanings and Design References. Emojipedia: Emoji Reference and Design Guide (2025)

Appendix

This project was developed independently as a hands-on exploration of LLMs for household inventory management. It offered valuable exposure to multiple technical areas and encouraged iterative design thinking throughout the development process.

Implementation Overview: I built the system from scratch, including the OCR pipeline with OpenCV preprocessing, Tesseract integration using multiple configurations, LLM prompt engineering for structured data extraction, Streamlit-based UI, and a JSON persistence layer. The modular design allowed for smooth testing and iterative improvements.

Challenges Faced:
OCR accuracy on real-world product images was lower than anticipated, necessitating extensive experimentation with image enhancement—this consumed a significant portion of development time. While LLM responses were generally consistent for English inputs, handling German product names and descriptions remained challenging despite prompt refinements. The system struggled with tokenisation and semantic interpretation of compound German words. Additionally, managing diverse date formats required robust regex design and thorough edge case handling.

Key Learnings: I gained practical experience with OCR beyond theory, improved my prompt engineering skills, and understood the importance of error handling in systems with noisy inputs. Iterative testing helped me appreciate user-centric design and system reliability.

Insights: Preprocessing quality is crucial for OCR performance—small tweaks made a big difference. LLMs proved effective in handling exceptions that rule-based systems struggle with. Simulating expert reasoning through prompts worked well, showing that complex architectures aren't always necessary. Persistent storage turned out to be vital for moving from prototype to usable tool.

Next Steps: I plan to explore fine-tuning for targeted extraction tasks, compare simulated vs. actual Mixture of Experts models, and experiment with computer vision for produce recognition. Optimising deployment for mobile platforms is also a priority to improve usability.

[1] Repository https://github.com/GrimGamer1999/smart-shelf-llm