NutriLensAI

Nutrient Analysis from Food Label Images Using Data-Driven Models

Developed by

- ARYAN NILESH SADVELKAR
- PRABIN RAJ SHRESTHA
- PEIYING CHEN

Introduction

With the rising trend of health consciousness, there is a growing demand for easily accessible, detailed nutritional information about food. However, obtaining this information often involves the tedious and time-consuming task of manually inputting ingredients and searching for their nutritional value. This project aims to address these challenges by automating the extraction of nutrient information from food labels and recipe images. By leveraging advanced artificial intelligence techniques, such as pre-trained models for image analysis and dietary restriction customization, the project seeks to provide users with accurate and tailored nutritional insights, empowering them to make better-informed food choices.

Background

The growing focus on health and well-being has led to an increased interest in understanding the nutritional content of foods consumed. Current methods for retrieving nutritional data often rely on manual processes, which can be inconvenient and prone to errors. Furthermore, individuals with specific dietary needs face additional challenges in identifying foods that align with their restrictions. Advances in artificial intelligence, particularly in computer vision and natural language processing, present an opportunity to address these issues by automating the extraction and interpretation of nutritional data. By combining pre-trained convolutional neural networks (CNN) for optical character recognition (OCR) and transformer models for nutrient information extraction, this project builds on the latest technological innovations to streamline this process.

The Problem

The manual process of obtaining nutritional information is tedious, inefficient, and challenging for individuals with dietary restrictions.

This project aims to automate the extraction and analysis of nutritional data from food labels and recipe images, providing accurate and customized insights to help users make informed dietary decisions.

The manual process of obtaining nutritional information involves identifying food ingredients, searching for their corresponding data in databases, and ensuring that they meet specific dietary requirements. This approach is both time-consuming and prone to inaccuracies. The challenge lies in automating this process to ensure accuracy and ease of use. By utilizing pre-trained models, the project aims to analyze images of food labels and recipes to extract ingredient information and provide a customized nutritional overview based on user-specific dietary restrictions. This innovation not only saves time but also enhances the accessibility of reliable nutritional data for diverse user needs.

Methodology

This project integrates multiple advanced technologies, including Optical Character Recognition (OCR), Natural Language Processing (NLP), Computer Vision, and Large Language Models (LLMs), to automate the extraction and analysis of nutritional information from food labels and recipe images.

Computer Vision

Computer Vision enables machines to interpret and understand visual information, such as images or videos. It utilizes techniques like image processing and deep learning to analyze visual data.The key techniques we covered used CRAFT also known as Character Region Awareness for Text Detection.

It is a state-of-the-art deep-learning model for text detection. Unlike traditional methods that detect words or lines, CRAFT focuses on detecting individual characters, offering improved accuracy in identifying text regions in complex layouts. Handles noisy backgrounds, diverse text styles, and curved baselines effectively.

Computer Vision helps segment and localize text regions from food labels and recipes, preparing the data for OCR processing. This step ensures that the extracted text is precise and ready for further NLP analysis.

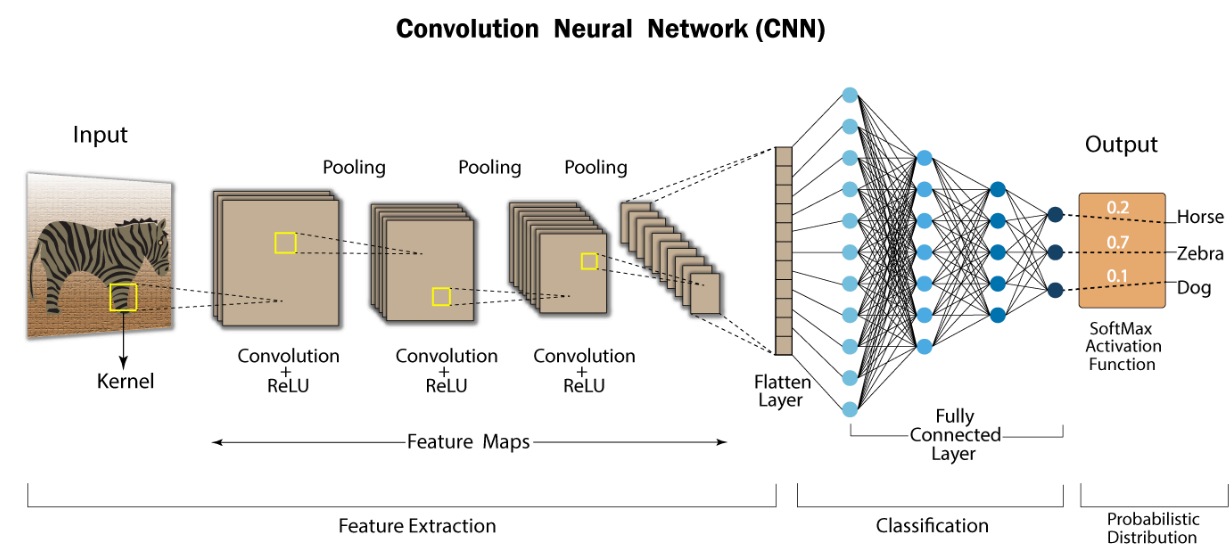
Optical Character Recognition (OCR)

Optical Character Recognition (OCR) is the technology used to convert text within images into machine-readable formats. In this project, OCR is critical for identifying and extracting textual information from food labels and recipe images.

We tried a variety of models, such as EAST and EasyOCR, and found that EasyOCR works best for our case. For the food labels we collected, EasyOCR produced highly acceptable results in recognizing the text. EasyOCR is a pre-trained CNN-based text recognition model used for its robustness and accuracy. It detects text regions in an image and processes them for further analysis. The extracted text regions are passed to a Convolutional Recurrent Neural Network (CRNN) for sequence modeling. CRNN is specifically designed to handle variable-length text sequences, converting image data into textual outputs efficiently

In this project OCR extracts essential text, such as ingredient lists and nutritional values, from complex backgrounds, curved text, or varied fonts, forming the foundation for subsequent analysis.

Convolution Neural Network (CNN)



A Convolutional Neural Network (CNN) is a type of deep learning model specifically designed to process and analyze data with a grid-like structure, such as images. CNNs are widely used in computer vision tasks, including image classification, object detection, and text recognition.

Text Detection

Before recognizing the text, EasyOCR needs to locate the regions in the image that contain text. CNNs are often used in text detection modules to:

- **Identify text regions:** CNNs scan the image to distinguish between text and non-text areas.
- **Generate bounding boxes:** The detected text regions are marked for further processing

RCNN

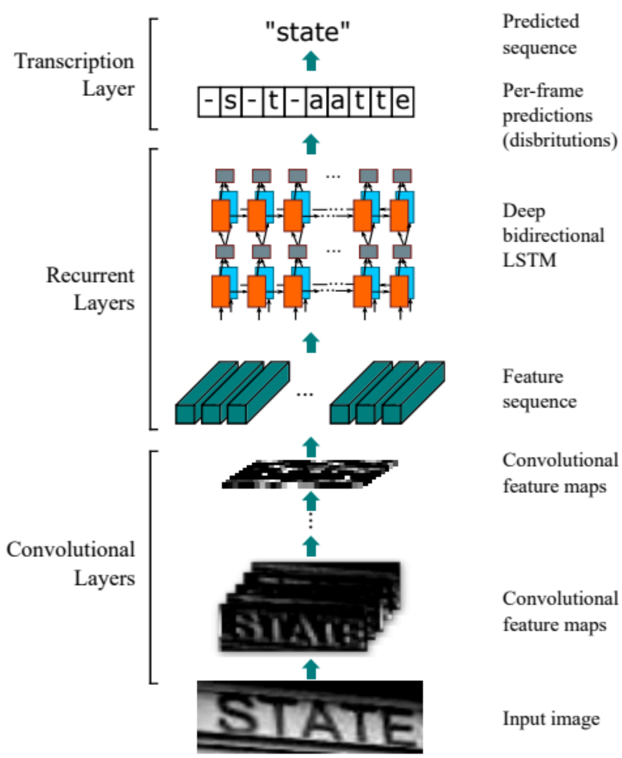


Figure 1. The network architecture. The architecture consists of three parts: 1) convolutional layers, which extract a feature sequence from the input image; 2) recurrent layers, which predict a label distribution for each frame; 3) transcription layer, which translates the per-frame predictions into the final label sequence.

RCNN (Region-based Convolutional Neural Network) plays an important role in EasyOCR's pipeline for text detection and recognition. EasyOCR does not explicitly use the classical RCNN architecture as it is primarily focused on text-related tasks, but it uses principles inspired by RCNN-like approaches. EasyOCR uses a BiLSTM-CNN architecture for text recognition, which is a combination of Convolutional Neural Networks (CNNs) and Bidirectional Long Short-Term Memory (BiLSTM) networks. This architecture effectively handles the challenges of recognizing text in images by leveraging the strengths of both CNNs for spatial feature extraction and BiLSTMs for sequence modeling.

The BiLSTM layer models the sequential nature of text:

- **Bidirectionality:** Unlike traditional LSTMs, BiLSTM processes input sequences from both directions (forward and backward). This helps the model understand context in both directions, crucial for recognizing characters that depend on surrounding characters for disambiguation.
- **Temporal Dependencies:** It captures relationships between characters, especially in cursive or overlapping text.

The input to the BiLSTM is the sequence of feature vectors produced by the CNN, and the output is a sequence of predictions for each feature vector.

BiLSTM-CNN Workflow in EasyOCR:

1. **Input Image:** A cropped image of the text is provided.
2. **CNN Feature Extraction:** The CNN extracts spatial features, converting the image into a sequence of feature vectors.
3. **BiLSTM Sequence Modeling:** The feature vectors are fed into a BiLSTM, which learns the context and dependencies between the features to recognize text sequences.
4. **CTC Loss for Training:** The output of the BiLSTM is compared with the ground truth text using CTC Loss, and the model is trained to minimize this loss.
5. **Text Prediction:** During inference, the output sequence is decoded into human-readable text.

Natural Language Processing (NLP)

NLP is a field of artificial intelligence that enables computers to understand, interpret, and generate human language in a meaningful way. It bridges the gap between human communication and machine understanding.

The key NLP processes in this project are:

- Text Classification: Identifies and categorizes specific components like ingredients and nutrient names.
- Entity Recognition: Recognizes entities such as ingredient names, nutrient quantities, and measurement units.
- Contextual Analysis: Processes and understands the relationships between entities, such as matching nutrients with their corresponding values.

After text extraction, NLP processes the raw text to identify and classify relevant information. For example, it distinguishes between ingredients and nutritional data, enabling accurate data organization and further analysis.

Tokenization

Tokenization is the process of splitting text into smaller units, such as words, subwords, or characters, called tokens. It helps in converting raw text into a format that can be processed by machine learning models.

Machines require structured input to process text. And, tokenization transforms unstructured text into numerical representations suitable for computation.

Named Entity Recognition (NER) and Part of Speech (POS) Tagging

The NER is used to Identify entities like names, dates, locations, etc., within the text. And POS Tagging identifies grammatical labels (e.g., noun, verb, adjective) to words in a sentence. The key purpose is to extract structured information from text and understand grammatical relationships.

Neural Network

A neural network is a series of connected layers of artificial neurons that process data to learn patterns and make predictions. This is the foundation for deep learning models like RNNs, LSTMs, and Transformers.

Neural Network includes the following components:

- Input Layer: Takes tokenized text as input.
- Hidden Layers: Extract features using weights and activations.
- Output Layer: Produces predictions or classifications.

Sequence-to-Sequence (Seq2Seq) Models

Seq2Seq models are a type of neural network architecture designed to map one sequence of data to another sequence of data. They are particularly effective for tasks where the input and output have different structures or lengths, such as language translation, text summarization, and question answering.

They are known to have two components; Encoder and Decoders. The encoder processes the input sequence and encodes it into a fixed-length representation, often referred to as a context vector or latent representation. Decoders, The output of the encoder consists of a context vector containing the essential information from the input sequence, regardless of its length. The decoder takes the context vector produced by the encoder and generates the output sequence one step at a time. At each step, the decoder predicts the next word in the output sequence based on the context vector and previously generated words.

Recurrent Neural Networks (RNNs)

RNNs are a type of neural network designed for sequential data, where the output of one step feeds into the next. They capture dependencies in sequences like sentences or time-series data.

RNN struggles with long sequences due to the vanishing gradient problem.

Long Short-Term Memory (LSTM)

LSTMs are a type of RNN that overcomes the vanishing gradient problem by introducing gated mechanisms (input, forget, and output gates) to control the flow of information. LSTM handles long-range dependencies better than standard RNNs. They are useful for tasks where earlier parts of the sequence significantly influence later parts (e.g., understanding context in long sentences).

Attention Mechanisms

Attention mechanisms are neural network components that help models focus on relevant parts of the input data. Instead of processing all tokens equally, attention assigns weights to emphasize relevant tokens. They are crucial for tasks where the importance of specific data varies across the input.

There are two Key Types of Attention:

Self-Attention: Enables models to compare each word in a sequence with all other words to capture relationships.

Contextual Attention: Focuses on relevant words or regions for a specific task.

Attention mechanisms improve the accuracy of text recognition in OCR models by focusing on critical regions of the input images. They are also used in transformers to enhance text understanding during the NLP phase.

Transformers

Transformers are advanced deep-learning architectures that have revolutionized natural language understanding. They use self-attention mechanisms to analyze input data and capture long-range dependencies between elements. Transformers play a pivotal role in processing extracted text, identifying patterns, and ensuring accurate contextual analysis. Their ability to handle sequential data with varying lengths makes them ideal for analyzing text data from food labels.

The Key components are:

- Encoder-Decoder Structure: Similar to Seq2Seq, with both encoding and decoding components.
- Self-Attention Mechanism: Captures relationships between all tokens in a sequence.
- Positional Encoding: Provides information about the order of tokens, as transformers process tokens in parallel.
- Feedforward Layers: Applies non-linear transformations after attention.

Transformers handle long-range dependencies effectively, and can Parallel processing enables faster training and inference. The greatest advantage is that transformers can use transfer learning.

Transfer learning

Transfer learning is a machine learning technique where a model trained on one task is reused as the starting point for a related but different task. Instead of training a model from scratch, transfer learning leverages the knowledge learned in a source task to improve performance or reduce the training time of a target task. transfer learning is one of the defining features that make transformer-based models like BERT, GPT, and T5 so powerful and versatile.

Transformers follow a two-step process that aligns perfectly with the principles of transfer learning.

1. Pretraining (Source Task):

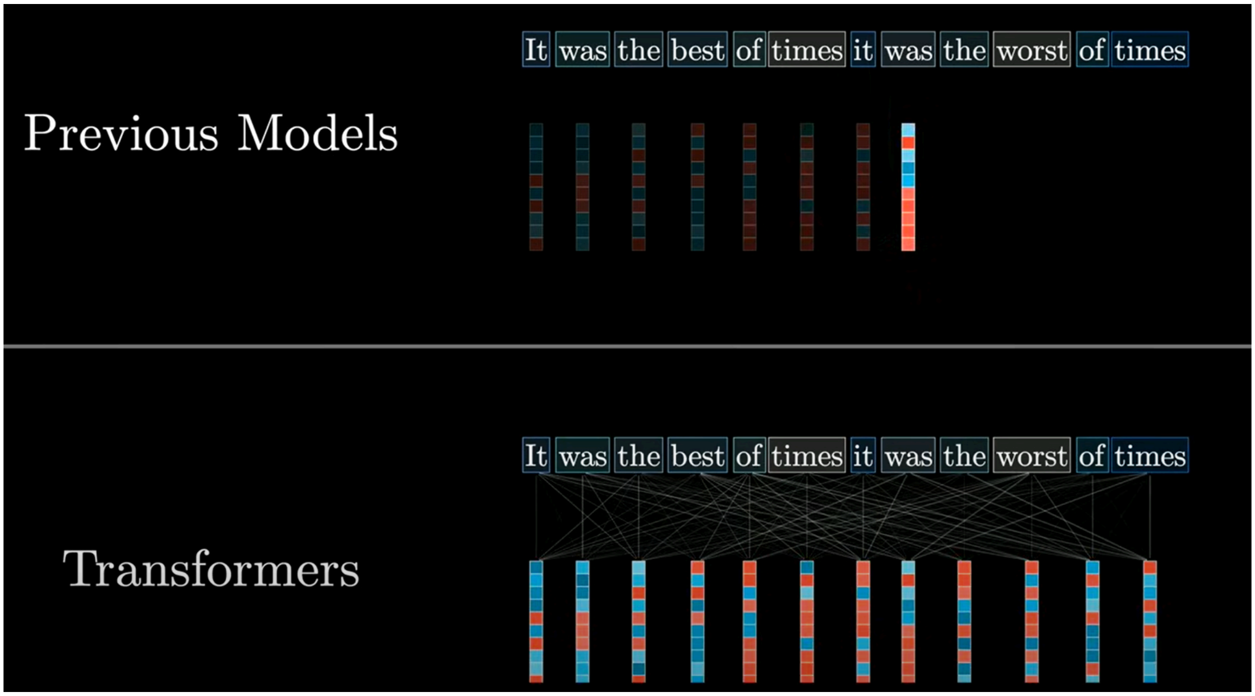
   The transformer model is trained on a large, general-purpose dataset. The task during pre-training is typically self-supervised, such as Masked Language Modeling (MLM), Causal Language Modeling (CLM), and Next Sentence Prediction (NSP).

2. Fine-Tuning (Target Task):

   In this step, the pre-trained transformer is then adapted to a specific task using a smaller dataset. The task can range from sentiment analysis and named entity recognition to machine translation and text generation.

   During fine-tuning, the model parameters are adjusted, but the general language understanding gained during pre-training is retained.

Transformers are highly effective at leveraging transfer learning. By pretraining on vast amounts of data, they develop a foundational understanding of language. This knowledge can then be transferred to specific tasks via fine-tuning, making transformers versatile, efficient, and powerful across diverse NLP applications.



Picture taken from 3Blue1Brown [YouTube video]

Large Language Models (LLM):

LLMs are advanced NLP models built on transformer architectures and trained on massive datasets to understand and generate human-like language.

Most LLM Models are trained on diverse text data to learn patterns and context. Using self-supervised learning objectives, such as predicting the next word in a sentence. These models are adapted to specific tasks using task-specific data. Through this training, the model learns grammar, facts about the world, and some reasoning abilities.

Once trained, LLMs can generate coherent and contextually relevant text by predicting subsequent tokens based on the input sequence. They can perform tasks like translation, summarization, and question-answering by leveraging the patterns learned during training.

LLMs generate text through a process called autoregressive generation, where each word is predicted based on the preceding context

Limitations of Large Language Models

Large Language Models (LLMs) have significantly advanced natural language processing, enabling applications like chatbots, translation services, and content generation. However, they possess inherent limitations that impact their reliability and effectiveness:

- **Bias and Fairness**

  LLMs can unintentionally generate biased content, reflecting prejudices present in their training data. This can lead to outputs that unfairly stereotype or misrepresent certain groups. For instance, studies have shown that LLMs may replicate gender biases in contexts like recommendation letters, potentially reinforcing harmful stereotypes.

- **Hallucination**

  LLMs sometimes produce information that is inaccurate or fabricated, a phenomenon known as “hallucination.” This occurs when models generate text that lacks factual grounding, leading to misleading or false outputs. For example, an LLM might assert incorrect historical facts or invent non-existent references, which can be problematic in applications requiring high accuracy.

- **Limited Context**

  Most LLMs have a fixed context window, meaning they can only consider a limited amount of prior text when generating responses. This constraint can cause models to lose track of earlier information in long documents or conversations, resulting in outputs that may lack coherence or relevance to the initial context

Approaches to solve the problem

To automate the extraction and analysis of nutritional information from food labels and recipe images, we propose a comprehensive process stack that integrates Optical Character Recognition (OCR), Natural Language Processing (NLP), and Large Language Models (LLMs). The workflow is as follows:

1. Image Acquisition: Capture high-quality images of food labels or recipes
2. Optical Character Recognition (OCR):
3. organizes extracted text
4. Incorporation of Dietary Constraints
5. Analysis Using Large Language Models (LLMs)
6. Prompt engineering and finetuning.

To ensure that LLMs produce appropriate and precise analyses, prompt engineering is applied. It is important to craft prompts that guide the LLM to focus on relevant aspects of the nutritional data and user constraints. This needs to be done iteratively, to improve the quality of the generated responses the prompts need to be continuously refined based on feedback.

The core technologies used:

- OCR converts images into machine-readable text.
- NLP processes and organizes extracted text.
- LLMs analyze and contextualize data for accurate nutrient extraction and provide recommendations.

Initial Approach

Our approach is to build a process stack leveraging existing or trained models. The process starts with an Image of the Food Label and nutrition, We would use OCR to extract the nutrition facts from the image. We would then include additional diet constraints and diet plans and feed them to LLM to get the nutrition analysis. We would perform prompt engineering to get the appropriate analysis.

The process begins with capturing images of food labels or recipes using standard imaging devices. OCR technology is employed to convert text within images into machine-readable text. This step involves:

1. Text Detection, where the goal is to Identify regions in the image that contain textual information.
2. Text Recognition: Converting the detected text regions into editable and searchable text formats.

Once the text is extracted, NLP techniques process and organize the information:

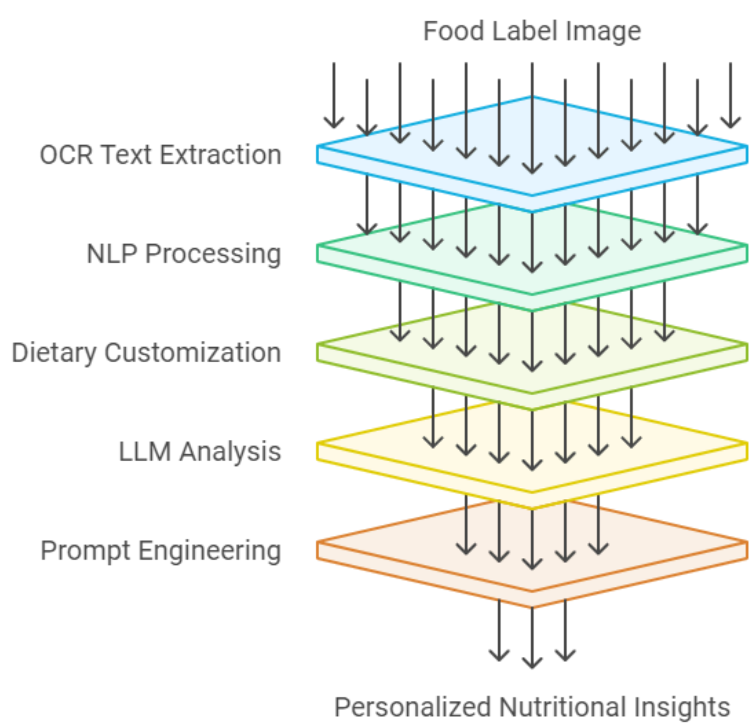
1. Text Segmentation: Dividing the continuous text into meaningful units such as words or sentences.

2. Entity Recognition: Identifying and classifying key elements like nutrients, ingredients, and quantities.
3. Data Structuring: Organizing the extracted information into structured formats (e.g., tables or databases) for easy analysis.

The next step in the stack is to incorporate User-specific dietary restrictions and preferences to tailor the nutritional analysis. This customization allows the system to provide personalized recommendations and insights. LLMs, such as OpenAI's GPT series, are employed to analyze the structured data. LLMs interpret the nutritional information within the context of user-specific dietary constraints. Then they provide insights into the nutritional content, potential dietary concerns, and suggestions for improvement.

To ensure that LLMs produce appropriate and precise analyses, prompt engineering is applied. It is important to craft prompts that guide the LLM to focus on relevant aspects of the nutritional data and user constraints. This needs to be done iteratively, to improve the quality of the generated responses the prompts need to be continuously refined based on feedback.

Nutritional Analysis Process



Challenges

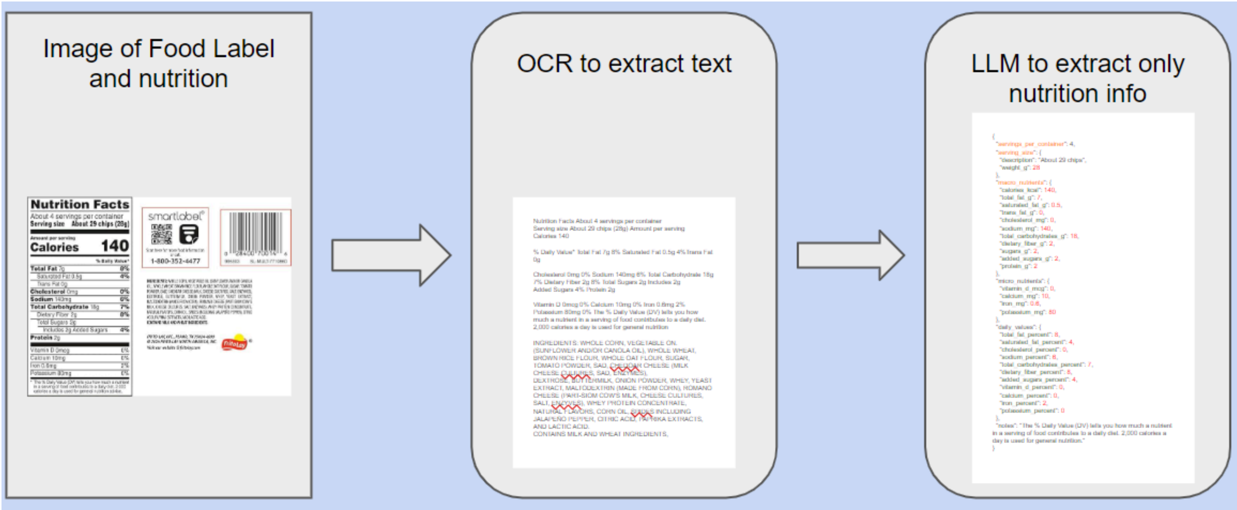
We faced several challenges,

- Locating Nutritional Facts: Identifying the precise location of nutritional information within images was difficult due to varying packaging layouts and the presence of extraneous marketing text.
- Isolating Relevant Details: The diversity in packaging designs and the inclusion of non-nutritional text made it challenging to extract only the pertinent nutritional facts.

Solution:

To address this issue, we employed a Large Language Model (LLM) to distinguish and extract relevant nutritional details from the OCR-processed text. This approach effectively standardized data formats, simplifying and streamlining the extraction process.

This methodology aligns with practices in the field, where LLMs are utilized to enhance information extraction from complex data sources. For instance, the Open Food Facts project has developed tools to detect and extract nutritional tables from images, addressing similar challenges in data extraction from diverse packaging formats



Final Approach:

The final approach integrates a stack of technologies and pretraind models to achieve a streamlined and efficient system for nutritional analysis. The steps include:

1. OCR Text Extraction:

   The process begins by employing Optical Character Recognition (OCR) to extract textual data from images of food labels or recipes. This step identifies and converts text regions into machine-readable formats, forming the foundation for further analysis.

2. Parsing Extracted Text with LLM:

   The extracted text is then parsed using a Large Language Model (LLM). The LLM processes and identifies the nutritional facts from the text and organizes the information into a structured JSON format, ensuring consistency and accessibility.

3. Incorporating Dietary Customization:

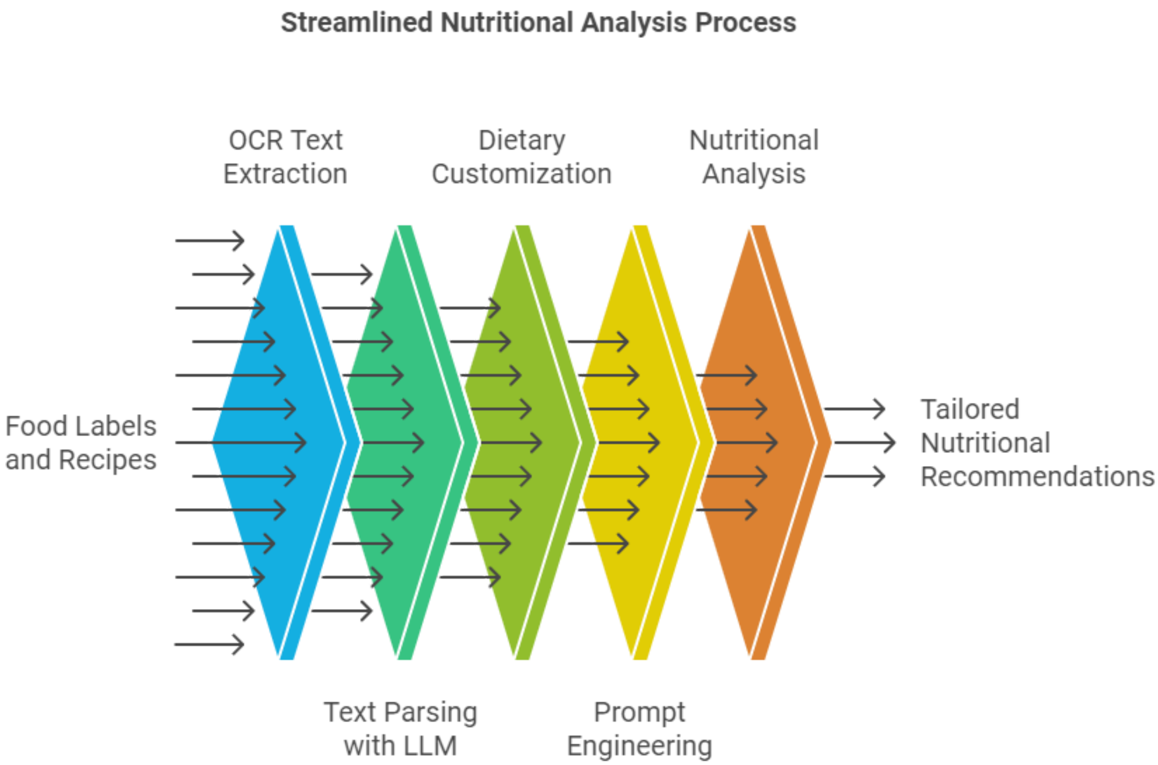
   User-specific dietary customizations, such as food allergies or personalized diet plans, are integrated into the system. This customization ensures that the analysis considers individual dietary restrictions and preferences.

4. Prompt Engineering:

   Effective prompts are crafted through prompt engineering to guide the LLM in performing the required tasks. This involves designing instructions that direct the model's focus on relevant aspects, such as dietary constraints and nutritional facts, ensuring precise and relevant outputs.

5. LLM-Driven Nutritional Analysis:

Finally, the LLM performs a comprehensive nutritional analysis based on the extracted and structured data, along with dietary customizations. The analysis provides insights into nutritional content, identifies potential dietary concerns, and offers recommendations tailored to the user.



Model Specifications

Model Exploration

We explored a variety of models to find the optimum model. An optimum model would give the best results for each task.

OCR

1. **EasyOCR**
    - **Description**:

      A ready-to-use OCR library supporting 80+ languages and popular writing scripts, including Latin, Chinese, Arabic, and Cyrillic.

    - **Advantages**:
        - Extensive language support.
        - Free and easy to use.
    - **Disadvantages**:
        - Limited ability to parse tables.
        - Does not retain text structure.
    - Repository: EasyOCR GitHub [https://github.com/JaidedAI/EasyOCR]
2. **Off-Nutrition-Table-Extractor**
    - **Description:**

      A pipeline designed for extracting nutritional tables, consisting of table detection, text detection, and OCR with post-processing. It utilizes the Single Shot Detector (SSD) model and TensorFlow's Object Detection API.

    - **Advantages:**
        - Tailored for nutrition tables.
        - integrating multiple processing steps.
    - **Disadvantage:**
        - Difficult to implement
    - **Repository**: Off-Nutrition-Table-Extractor GitHub
        - [https://github.com/openfoodfacts/off-nutrition-table-extractor]
3. **Microsoft OmniParser**
    - **Description**:

      A general screen parsing tool that converts UI screenshots into structured formats, enhancing LLM-based UI agents.

    - **Disadvantage**
        - Not tailored to read Nutrition labels
        - Requires additional tuning, and programming

- **Repository:** [https://huggingface.co/microsoft/OmniParser]

LLMS

OpenAI GPT-4o

- **Architecture and Parameters:** GPT-4 is reported to utilize a Mixture of Experts (MoE) architecture, comprising eight models, each with 220 billion parameters, totaling approximately 1.76 trillion parameters.
- **Training Data:** The model was trained on a dataset encompassing around 13 trillion tokens, sourced from a diverse array of publicly available text.
- **Advantages**:
  - Highly powerful and state-of-the-art.
- **Disadvantages**:
  - Requires token purchasing, making it cost-prohibitive for extensive use.

Llama Models

- Llama-3.2-1B-Instruct
  - **Parameters:** 1 billion parameters.
  - **Context Length:** Supports up to 128,000 tokens, allowing for processing of long sequences.
  - **Training Data:** Pretrained on up to 9 trillion tokens from publicly available sources.
  - **Advantages:** Lightweight, free, and supports local processing.
  - **Disadvantages:** Limited input/output token size.
- Llama-3.2-3B-Instruct
  - **Parameters:** 3 billion parameters.
  - **Context Length:** Supports up to 128,000 tokens
  - **Training Data:** Pretrained on up to 9 trillion tokens from publicly available sources.
  - **Advantages:** Free and supports local processing, Balances performance and resource requirements, offering improved capabilities over the 1B model.
  - **Disadvantages:** Limited input/output token size.

Final Model

Based on performance and project requirements, the following models were chosen:

OCR Model:
EasyOCR
- Selected for its simplicity and effectiveness in recognizing text from diverse languages and scripts.

LLM Model:
Llama-3.2-1B-Instruct

- Chosen for its lightweight nature, free availability, and ability to perform local processing, ensuring cost efficiency and ease of deployment.

Libraries Used

This project utilizes a variety of libraries to streamline the development process and ensure efficient implementation of the required functionality.

1. EasyOCR

- A ready-to-use OCR library that supports over 80 languages and various writing scripts, including Latin, Chinese, Arabic, Devanagari, and Cyrillic.
- Used for extracting text from food labels and recipe images.
- Features:
  - Extensive language support.
  - Simple integration for OCR tasks.
- Documentation: EasyOCR on PyPI [https://pypi.org/project/easyocr/]

2. Torch (PyTorch)

- A Python library that provides:
  - Tensor computation with GPU acceleration.
  - Support for building and training deep neural networks using an autograd system.
- Provides the foundational framework for machine learning tasks and computational operations.
- Features:
  - Efficient tensor computation.
  - Strong GPU support for deep learning models.
- Documentation: Torch on PyPI [https://pypi.org/project/torch/]

3. Transformers

- A library offering thousands of pre-trained models for tasks involving text, vision, and audio.
- Used to integrate transformer-based models for tasks like text parsing and analysis.
- Features:
  - Access to pre-trained models like GPT, BERT, and others.
  - Support for multi-modal tasks.

- Documentation: Transformers on PyPI [https://pypi.org/project/transformers/]

4. Json

- A library for serializing and deserializing Python objects into JSON-compatible formats.
- Converts extracted and processed data into JSON format for structured analysis and further processing.
- Features:

    - Converts Python objects to JSON strings or dictionaries.
    - Customizable and extendable.

- Documentation: [Jsons on PyPI] [https://pypi.org/project/jsons/]

5. Aisuite

- A unified interface for interacting with multiple generative AI providers.
- Allows seamless integration and testing of different LLM providers without requiring code changes.
- Features:
    - Simplifies swapping between AI providers.
    - Streamlines experimentation with generative AI models.

- Documentation: [Aisuite on PyPI] [https://pypi.org/project/aisuite/]

Implementation:

The process stack for the project has been implemented using Google Colab, providing a cloud-based environment for executing the code and integrating the various components.

Link to Implementation: Google Colab Implementation

https://colab.research.google.com/drive/1hssG0A63Darm7yq_Ort5ud755MzsZQEL?usp=sharing

Results

Below are the results from processing.

Nutrition Facts	
1 serving per container	
Serving size	**1 Bag (42g)**
Amount per serving	
Calories	**230**
	% Daily Value*
Total Fat 14g	**18%**
Saturated Fat 1.5g	**8%**
Trans Fat 0g	
Polyunsaturated Fat 1.5g	
Monounsaturated Fat 11g	
Cholesterol 0mg	**0%**
Sodium 260mg	**11%**
Total Carbohydrate 23g	**8%**
Dietary Fiber 3g	**11%**
Total Sugars 1g	
Includes 1g Added Sugars	**2%**
Protein 3g	
Vitamin D 0mcg 0% •	Calcium 0mg 0%
Iron 1mg 6%	• Potas. 483mg 10%

* The % Daily Value (DV) tells you how much a nutrient in a serving of food contributes to a daily diet. 2,000 calories a day is used for general nutrition advice.

Nutrition Analysis:
Nutritional Analysis:
This ready-to-eat food product contains 230 calories per serving, with no fat or protein but a significant amount of sugar (24g, including added sugars). The high sugar content makes it less ideal for regular consumption, especially for individuals managing blood sugar levels or aiming for weight control.

Positive Aspects:

* The product provides a small amount of vitamin C (30mg) per serving, which is a positive aspect.

Negative Aspects:

* The high sugar content (24g, including added sugars) is concerning, as it may lead to an excessive intake of sugar, potentially contributing to various health issues such as obesity, type 2 diabetes, and dental cavities.
* The product is high in saturated fat (1.5g) and sodium (260mg), which may increase the risk of heart disease and high blood pressure.
* The lack of beneficial nutrients like fiber, vitamins, and minerals is notable, particularly the absence of vitamin D, calcium, iron, and vitamin B12.
* The product contains a significant amount of added sugars (1g), which is a concern for those managing their sugar intake.

Concerning Ingredients:

* The presence of added sugars (24g) is a major concern, as excessive sugar consumption has been linked

Nutritional Analysis:

This ready-to-eat food product contains 100 calories per serving with no fat or protein but a significant amount of sugar (24g, including added sugars). While it provides 30mg of vitamin C (a positive aspect), the high sugar content makes it less ideal for regular consumption, especially for individuals managing blood sugar levels or aiming for weight control.

Positive Aspects:

* The product contains a small amount of vitamin C, which is beneficial

```
for overall health.
* The sugar content is relatively low compared to other snack foods.

Negative Aspects:

* The high sugar content (24g) exceeds the daily recommended intake for
most adults.
* The product contains no fat or protein, making it unsuitable for those
looking to increase their intake of these essential nutrients.
* The presence of artificial flavors, colors, and preservatives may be
detrimental to overall health.

Concerning Ingredients:

* The product contains a high amount of added sugars, which can lead to
a rapid spike in blood sugar levels and contribute to chronic diseases.
* The presence of artificial flavors, colors, and preservatives may be
detrimental to overall health.

Suggestions:

* Consider replacing the high-sugar content with natural sweeteners or
reducing the amount of added sugars.
* Look for products with fewer ingredients
```

Smooth Vanilla

Nutrition Facts

1 serving per container
Serving size 12 fl oz (355mL)

Amount per serving	
Calories	**180**
	% Daily Value*
Total Fat 7g	**9%**
Saturated Fat 1g	**5%**
Trans Fat 0g	
Cholesterol 0mg	**0%**
Sodium 270mg	**12%**
Total Carb 8g	**3%**
Dietary Fiber 3g	**11%**
Total Sugars 4g	
Inc. 4g Added Sugars	**8%**
Protein 20g	**24%**
Vitamin D 0mcg	0%
Calcium 40mg	3%
Iron 4mg	20%
Potassium 50mg	1%

* The % Daily Value (DV) tells you how much a nutrient in a serving of food contributes to a daily diet. 2,000 calories a day is used for general nutrition advice.

Ingredients: Water, OWYN™ Protein Blend (Pea Protein, Organic Pumpkin Seed Protein, Organic Flax Oil), Contains 2% or less of the following: Soluble Fiber, Organic Cane Sugar, Sunflower Oil and/or Safflower Oil, Natural Flavors, Sunflower Lecithin, Greens Blend (Broccoli, Spinach, Kale) Monk Fruit Extract, Himalayan Pink Salt, Guar Gum, Gellan Gum.

100% gluten free

100% vegan

NON GMO Project VERIFIED nongmoproject.org

allergy friendly no dairy, soy, peanut, tree nuts, sesame, wheat/gluten

UDE bpa free

```
Nutritional Analysis:

This food item contains 100 calories per serving. It is a ready-to-eat
food product with no fat or protein but a significant amount of sugar
(24g, including added sugars). The nutritional information is as
follows:

- Calories: 100
- Sugar: 24g
- Saturated Fat: 0g
- Sodium: 50mg
- Fiber: 0g
```

```
- Vitamin C: 30mg
- Calcium: 0mg
- Iron: 0mg

Positive Aspects:

- The high sugar content is a significant concern, as excessive sugar
consumption is linked to various health issues, including obesity, type
2 diabetes, and dental caries.
- The lack of essential nutrients like fiber, vitamins, and minerals is
also notable, as these nutrients are crucial for maintaining optimal
health.

Negative Aspects:

- The absence of essential nutrients like fiber, vitamins, and minerals
may lead to nutrient deficiencies over time.
- The high sugar content may not be suitable for individuals with
diabetes or those who are trying to manage their blood sugar levels.
- The lack of protein and healthy fats may make it challenging for
individuals who require these nutrients for optimal health.

Concerning Ingredients:

- The absence of fiber and essential nutrients
```

Nutritional Analysis:

This food item contains 20 calories per serving, with no fat, protein, or significant amounts of sugar (only 3g of added sugars). However, it provides a considerable amount of dietary fiber (0g) and some beneficial vitamins and minerals (such as vitamin C and calcium). The high sugar content is a concern, especially for individuals managing blood sugar levels or aiming for weight control.

Positive Aspects:

* The high dietary fiber content may help with satiety and digestion.

```
* The presence of vitamin C is a positive aspect, as it supports immune
function and overall health.
* The food item is relatively low in saturated fat and cholesterol,
which may be beneficial for heart health.

Negative Aspects:

* The high sugar content is a significant concern, as excessive sugar
consumption has been linked to various health problems, including
obesity, type 2 diabetes, and heart disease.
* The lack of beneficial vitamins and minerals (such as vitamin E and
thiamin) may be a drawback for individuals with specific dietary needs
or restrictions.
* The absence of protein and other essential nutrients like calcium and
iron may be a limitation for individuals with certain dietary
requirements.
```

Conclusion

This project demonstrates a robust and innovative approach to automating the extraction, analysis, and contextualization of nutritional information from food labels and recipe images. By leveraging advanced technologies such as Optical Character Recognition (OCR), Natural Language Processing (NLP), and Large Language Models (LLMs), we have created a streamlined process stack capable of addressing diverse user needs, including dietary restrictions and personalized nutrition analysis.

The integration of EasyOCR for efficient text extraction, combined with the lightweight and locally deployable Llama-3.2-1B-Instruct model, ensures a cost-effective and scalable solution. The use of prompt engineering further enhances the precision and relevance of the outputs, while JSON formatting facilitates structured data handling for downstream applications.

Despite challenges such as inconsistent text layouts and the need to isolate relevant details from extraneous content, our solution of incorporating LLMs for precise parsing and analysis, effectively addressed these issues. The project not only simplifies the process of accessing nutritional information but also empowers users to make informed dietary decisions based on accurate and customized insights.

Overall, this project underscores the potential of AI-driven systems to revolutionize the way we interact with and interpret nutritional data in our daily lives.

Future Work

Future work could focus on expanding the model’s capabilities, such as improving table parsing, accommodating multilingual datasets, and optimizing for real-time applications. To enhance the current system and expand its usability, the following areas of development are proposed:

Build a Mobile or Web Application:

Developing a user-friendly application would enable broader accessibility and seamless interaction with the system. A dedicated app could enable users to capture food label images directly, setup input dietary preferences, and instantly receive personalized nutritional analyses.

Fine-Tune Prompts:

Continued refinement of prompt engineering is essential to further improve the precision and contextual accuracy of the Large Language Model (LLM) outputs. Iterative testing and optimization of prompts will ensure the system consistently provides high-quality, relevant responses tailored to user queries.

Incorporate Comprehensive Diet Plans:

Integrating diet plan recommendations into the system would provide users with actionable insights based on their dietary goals. By linking nutritional analysis to pre-defined or customizable diet plans, the system could suggest meals, track caloric intake, and support long-term health management.

These advancements will transform the system into a comprehensive nutritional assistant, offering enhanced functionality and value to users while maintaining the core strengths of the existing process stack.

Information for Each Contributors

ARYAN NILESH SADVELKAR

SU Email Address: asadvelk@syr.edu

Phone Number: 680 207 9488

PRABIN RAJ SHRESTHA

SU Email Address: pshres01@syr.edu

Phone Number: 315 952 9863

Website: prbn.info

PEIYING CHEN

SU Email Address: pchen21@syr.edu

Phone Number: 680 910 1753

References

- OCR:https://github.com/clovaai/deep-text-recognition-benchmark
- Transformer: Vaswani, A. "Attention is all you need." Advances in Neural Information Processing Systems (2017).
- 3Blue1Brown "Transformers (how LLMs work) explained visually" Youtube https://www.youtube.com/watch?v=wjZofJX0v4M&t=1173s&ab_channel=3Blue1Brown
- Andrej Karpathy "Let's build GPT: from scratch, in code, spelled out." Youtube https://www.youtube.com/watch?v=kCc8FmEb1nY&ab_channel=AndrejKarpathy
- OpenAI, "GPT-4 Architecture, Datasets, Costs, and More," SemiAnalysis, Jul. 2023. https://semianalysis.com/2023/07/10/gpt-4-architecture-infrastructure.
- J. Ramaswamy, "The Next Era of AI: Inside the Breakthrough GPT-4 Model," Hackernoon. https://hackernoon.com/the-next-era-of-ai-inside-the-breakthrough-gpt-4-model.
- Originality.ai, "GPT-4 Statistics," Originality.ai. https://originality.ai/blog/gpt-4-statistics
- Meta, "Llama 3.2-1B-Instruct: Model Card," and "Llama 3.2-3B-Instruct: Model Card," GitHub. https://github.com/meta-llama/llama-models/blob/main/models/llama3_2/MODEL_CARD.md