College of Health Solutions
Arizona State University

Embedded Machine Intelligence Lab

NutriGen: Personalized Meal Plan Generator Leveraging Large Language Models to Enhance Dietary and Nutritional Adherence

Saman Khamesian*, Asiful Arefeen, Stephanie M. Carpenter, Hassan Ghasemzadeh

Embedded Machine Intelligence Lab
Arizona State University

ghasemzadeh.com

47th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)

Agenda

- Introduction
- Methodology
  - Input Data
  - Prompt Engineering
- Experiments
  - Dataset
  - Selected Models
- Results
- Limitations and Future Works

Introduction

- Many people struggle with healthy eating.
- **Challenges:**
  - ✓ Calculating calories and nutrients
  - ✓ Planning balanced meals every day
  - ✓ Lack of nutrition knowledge
  - ✓ Finding recipes for available ingredients

- These barriers contribute to **poor long-term diet** adherence.

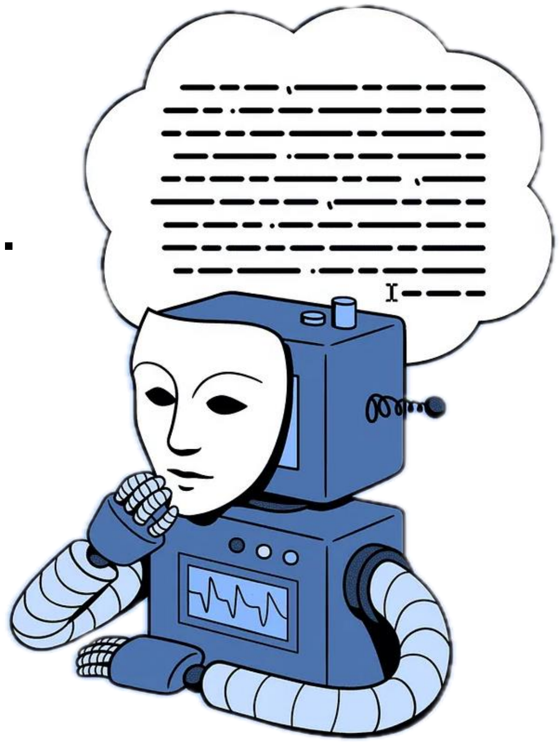
Introduction

- **Limitations of Existing Systems:**

  - ✓ Require too much user input.
  - ✓ Don’t support food culture.
  - ✓ Don’t adapt to ingredient availability.
  - ✓ Often recommend single food items, not full meals.
  - ✓ Lack of scalability or practicality for everyday use.

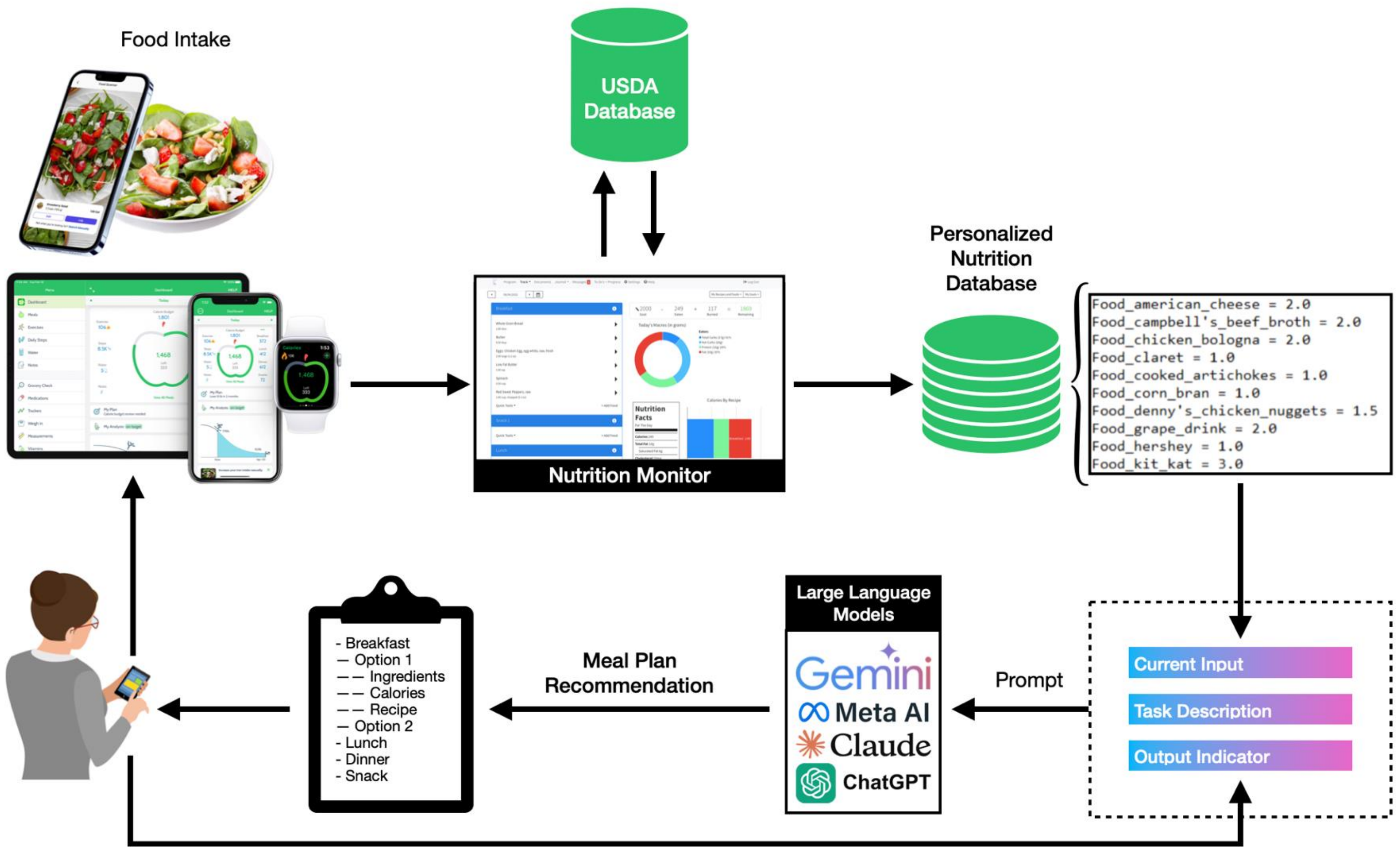
Introduction

- **What is NutriGen?**
  - A system that generates **personalized meal plans**.
  - Built using **large language models (LLMs)**.
  - Makes recommendations based on your:
    - Preferences
    - Restrictions
    - and calorie goals.



Large Language Model?
A deep learning model trained on vast text data to understand and generate human-like language. It can summarize, translate, answer questions, and create text.

Methodology



How NutriGen Works?

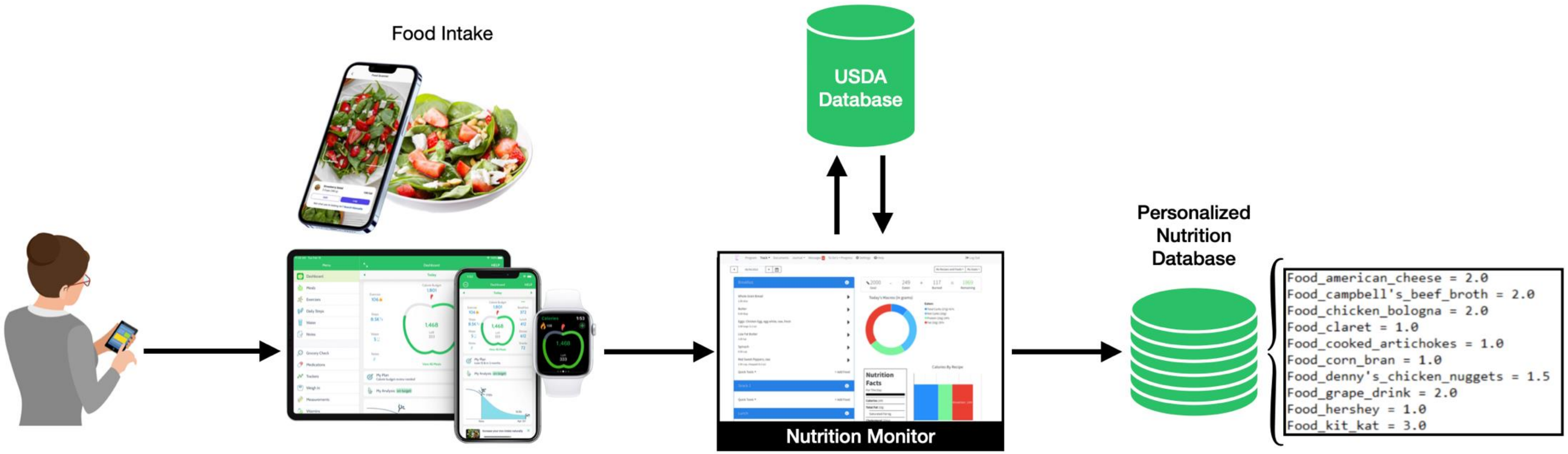
1) Collects input from users (food habits, goals, restrictions).

2) Builds a personalized nutrition database.

3) Uses LLMs to generate complete meal plans.

4) Includes recipes, portion sizes, and calorie counts.

Input Data Collection



- Accepts data via:
  - Image-based food logging (with ML and OCR).
  - Manual text or voice input (with NLP).
  - Third-Party applications (MyFitnessPal)
- Combines user input with trusted sources (e.g., USDA database).

Prompt Engineering

- Once the personalized nutrition database is ready, we construct a structured prompt that guides the language model to generate targeted meal plans.

$$P = \{I_{current}, T_{task}, O_{output}\}$$

User's dietary profile:
- Food Intake history,
- Preferences,
- Constraints.

Task instruction:
- Generate meal plans
- Matching calorie/macronutrient targets)

Desired output format:
- Structured meal plans

Prompt Engineering

User's goal is to create a meal plan with:

- Total calories: `{total_calories}` kcal.
- Total protein: `{target_protein}` g.
- Total sugar: `{target_sugar}` g.

Input:

Calories/Micronutrition's Targets

It can also include preferences/constraints.

The plan must include:

- Breakfast, lunch, dinner, and snacks.
- The calorie count for each meal (e.g., "Breakfast: 400 kcal").
- At the end of each meal plan option, provide the total calories, total fat, total protein, and total carbohydrate.
- For each meal component, specify portion sizes, including the number of items or volume (e.g., "1 Kit Kat bar (45g)," "1 hamburger with a 150g beef patty, bun, and lettuce").
- Provide a short recipe for each item, detailing how it can be prepared (e.g., "Grill the patty for 5 minutes, then assemble with lettuce, tomato, and a bun").

Task Instruction

Provide three different meal plan options for diversity.

- Use familiar dishes instead of listing individual food items. For example, use "hamburger" instead of "150 grams of meat with bun and lettuce."
- Ensure the plan adheres to the user's preferences and restrictions and meets the specified targets while maintaining a balanced nutritional profile.

Output Format

Here are the available items: `{menu_input}`

Input: *Food Intake history*

Experimental Setup

- We designed a system to generate 10 diverse and plausible daily food intake profiles using a set of 200 randomly selected meals from the USDA dataset.

```
Food_Barbequeue_Lays = 1.0
Food_Garden_Pizza = 1.0
Food_Milano_double_chocolate = 1.0
Food_baked_cheddar_ruffles = 1.0
Food_beef_angus_burger_patty = 1.0
Food_chocolate_milkshake = 1.0
Food_eggs_benedict = 0.5
Food_tortilla_chips = 1.0
-----------------------------------
calories = 1573.25
protein = 54.0
sugar = 58.3
```

This box presents a sample nutrition profile with predefined food items and nutritional targets.

Implementation and Dataset are available at https://github.com/SamanKhamesian/NutriGen

Experimental Setup

- In our experimental evaluations, we selected several advanced language models from leading organizations:

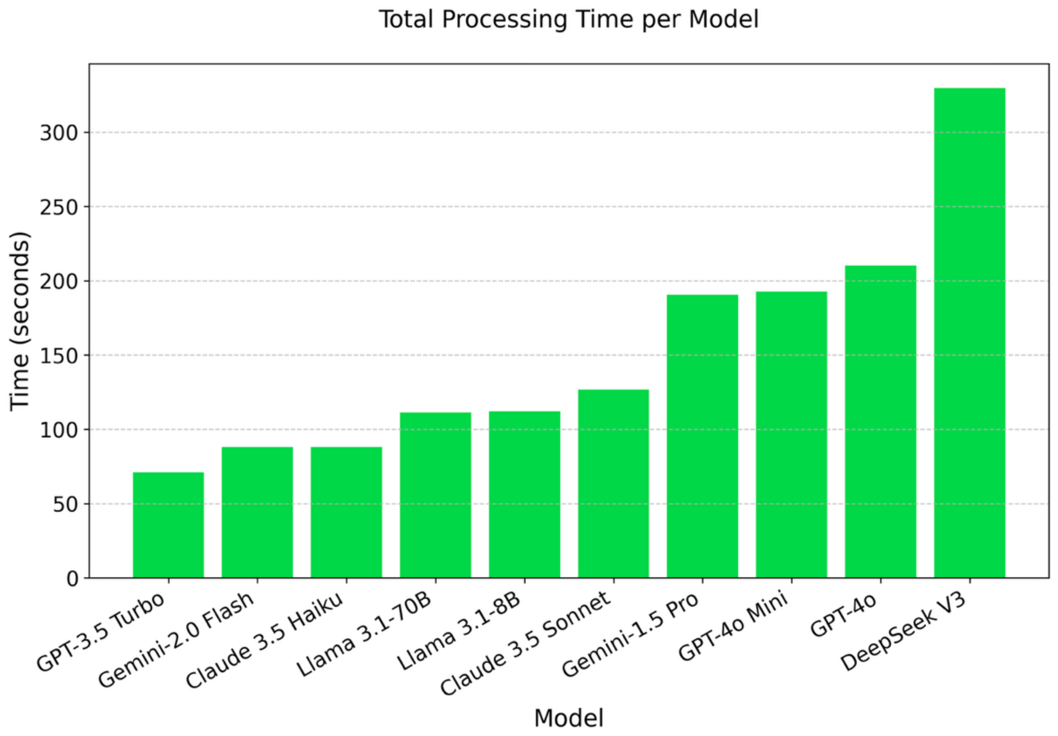
OpenAI	Google Gemini	Anthropic	Meta
GPT-4o GPT-4o Mini GPT-3.5 Turbo	Gemini 2.0 Flash Exp Gemini 1.5 Pro	Claude 3.5 Sonnet Claude 3.5 Haiku	Llama 3.1-70B Llama 3.1-8B

and DeepSeek-V3

Results

Processing Time Comparison

- To evaluate the computational efficiency of each model, we measured the total processing time required to generate 10 outputs.
- As expected, smaller and optimized models such as **GPT-3.5 Turbo**, **Gemini 2.0 Flash**, and **Claude 3.5 Haiku** demonstrated the fastest processing times.



Results

Reported Nutritional Values vs. USDA Dataset

	Input 1	Input 2	Input 3	Input 4	Input 5	Input 6	Input 7	Input 8	Input 9	Input 10
Claude 3.5 Haiku	570.69	124	88	103.5	-	75	76.75	100	83	113
Claude 3.5 Sonnet	138.09	124	88	105	96	75	77	96	83	113
DeepSeek V3 [22]	138.09	96.6	88	-	-	-	-	100	82.84	113
Gemini 1.5 Pro [23]	588	173.59	87.75	153.5	445.5	75	76.75	96	83	52.59
Gemini 2 Flash [23]	589.33	173.59	87.75	153.5	445.5	75	103.41	96	83	48
GPT-3.5 Turbo [25]	138.09	73.59	37.75	53.5	**45.5**	**25**	**26.75**	46	**32.84**	62.59
GPT-4o [24]	50	**24**	**38**	**45**	563	41.6	50	**36**	74.6	52.59
GPT-4o Mini	60.03	27.5	50	50	161.66	55	**26.75**	47.33	35	53.33
Llama 3.1 8B [21]	88.09	58.33	50	60	46	58.33	**26.75**	-	50	**43.33**
Llama 3.1 70B [21]	**45**	60	50	50	46.66	40	48.33	46	50	63

- We examined the accuracy of the nutrition facts in the generated meal plans. Each model was tasked with generating 3 meal plans per input.
- We calculated the **Mean Absolute Error (MAE)** between reported total calories per input and USDA reference values.

Results

Adherence of Meal Plans to User-Specified Targets

- We evaluated how closely each model's meal plans matched user-specified calorie targets.
- For each input, we compared the average total calories of generated meal plans to the defined target value.

Parameter	Description
P	Items per meal plan
N = 10	number of input profiles
M = 3	meal plans per input
$C_{actual\,(i,j,k)}$	calorie content of item k in meal plan j for input i
$C_{target,\,i}$	target calories for input i

Model	MAE	MAE (%)
Claude 3.5 Haiku	128.23	8.99
Claude 3.5 Sonnet	99.16	4.85
DeepSeek V3 [22]	190.61	4.85
Gemini 1.5 Pro [23]	182.16	10.44
Gemini 2 Flash [23]	179.16	9.74
GPT-3.5 Turbo [25]	54.16	3.68
GPT-4o [24]	189.76	13.47
GPT-4o Mini	329.06	24.67
Llama 3.1 8B [21]	**34.14**	**1.55**
Llama 3.1 70B [21]	109.21	8.08

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{1}{M}\sum_{j=1}^{M}\sum_{k=1}^{P} c_{\text{actual},i,j,k} - C_{\text{target},i}\right|$$

Limitations and Future Works

Current Limitations:

- Output token limits caused incomplete meal plans
- Calorie estimates were sometimes inconsistent
- Simulated user data may limit generalizability

Future Directions:

- Integrate a ChatBot for interactive, user-driven updates
- Use multimodal LLMs for food image analysis.
- Support multiple languages

ASU College of Health Solutions
Arizona State University

EMBEDDED MACHINE INTELLIGENCE LAB

Thank You!

Saman Khamesian
s.khamesian@asu.edu
https://github.com/SamanKhamesian

Hassan Ghasemzadeh
hassan.ghasemzadeh@asu.edu
https://ghasemzadeh.com/