Visual Question Answering for Bioavailable Iron: Food Image Analysis and Component Estimation using Small Vision-Language Models

Chelsea Ramos
University of Texas at Austin
chelseanbr@utexas.edu

Abstract

Food image analysis with dish feature identification and nutrient estimation has become more prominent in research and in our daily lives. Accurate predictions are critical for improving our ability to monitor diet and nutrition, especially for alleviating iron deficiencies. Current systems only estimate total iron rather than bioavailable iron or its individual factors, such as ingredient portions, cooking method, and iron-inhibiting micronutrients like calcium. We address this gap by creating a new Visual Question Answering (VQA) dataset with a subset of MM-Food-100K that we supplemented with iron, calcium, and vitamin C measurements from the USDA FoodData Central API. We also finetune small Vision-Language Models (VLMs) from the SmolVLM family using efficient LoRA (Low-Rank Adaptation) techniques. Our final finetuned 2.2B SmolVLM-Instruct model achieved a total classification accuracy of 0.587 and a Mean Absolute Error (MAE) of 1.45 mg for iron, demonstrating its potential and establishing a baseline for using VLMs to predict key components from real-world images that are necessary for estimating and calculating bioavailable iron calculations.

1 Introduction

Iron deficiency, with or without anemia, has been estimated to affect more than 3 billion people worldwide and cause debilitating symptoms such as fatigue and exercise intolerance (Al-Naseem et al., 2021). Although we can obtain iron through our diet, the total iron present in food differs from the amount of iron the body can absorb and use, which is called bioavailable iron (Hallberg and Hulthén, 2000). Consequently, the amount of absorbed iron is significantly lower than the total iron listed on nutrition labels and in nutritional data sources. In addition, different ingredients can influence the absorption of iron within a dish. For example, calcium sources such as cheese can inhibit iron absorption, while vitamin C sources such as bell peppers can enhance it (Hurrell and Egli, 2010).

One way we can achieve adequate nutrition is by tracking our diet with mobile apps that can now even analyze food images to estimate the nutritional profiles of dishes. However, most food image analysis models and apps provide only estimated measurements of calories and macronutrients such as fat, protein, and carbohydrates, but not micronutrients such as iron. Those that provide iron estimates have measurements based on total iron, not bioavailable iron. Bioavailable iron provides a better estimate because it considers whether iron absorption inhibitors, such as cheese or cream, or enhancers, such as peppers or broccoli, are also present in the food.

Figure 1: An image of pho, a Vietnamese noodle dish, from the MM-Food-100K dataset (Dong et al., 2025)

We aim to bridge the gap by fine-tuning a vision language model (VLM) to analyze food images and predict estimates of iron, calcium, and vitamin C micronutrients for bioavailable iron calculations. We will do so through Visual Question Answering (VQA), in which we aim to train a model to provide accurate natural language answers to natural language questions about the provided visual content

(Antol et al., 2015). This effort involves creating a novel VQA dataset, specifically designed to estimate the micronutrients necessary for calculating bioavailable iron, which is currently a gap in food image analysis systems.

Specifically, we created Question-Answer (QA) pairs that will allow the model to predict helpful information for determining bioavailable iron, such as dish name, food category, and cooking method, ingredient portions, and our target micronutrient estimates, as well as other nutritional details for comparison.

Phytates and polyphenols are also recognized as significant inhibitors of iron absorption, as noted by Hurrell and Egli (2010). However, these compounds were not easily or reliably accessible through the USDA FoodData Central API or any other available APIs or datasets. As a result, we decided to exclude them and forego calculating a single measurement of bioavailable iron for each data point. Our aim continued to be centered on analyzing food images and predicting most of the individual components necessary for determining bioavailable iron through the data we could access.

2 Related Work

2.1 Food Image Analysis

Recent advances in deep learning and mobile technology have enabled us to perform food analysis and dietary planning without having to be professional nutritionists (Niu et al., 2024). Using our smartphones, we can capture photos of our meals and use an app to obtain nutritional estimates, but there are still certain limitations, with ongoing research to address these challenges. For example, a major factor in nutrition is the cooking style or method, which has been a difficult task for both labelers and VLMs to determine based on a single image (Romero-Tapiador et al., 2025).

Other research in the area includes fine-grained food analysis for better accuracy. He et al. (2013) proposed a way to perform food image analysis through image segmentation and identification to estimate portion weights of different foods within a single image. He et al. (2014) later extended their work to improve food classification since foods like brownies and chocolate cake look quite similar, so they investigated deeper into "color, texture, and local region features." As evidenced by these studies, food image analysis is a complicated task due to many factors involved.

2.2 Nutrient Estimation

The area of deep learning for nutrient estimation has seen many significant advancements. One strategy for estimating nutrients, presented by Jiang et al. (2020), involved the use of deep Convolutional Neural Network (CNN) models. However, Vision Transformers have demonstrated superior performance over CNNs, as evidenced by the findings of Banerjee et al. (2024). Furthermore, Cheng et al. (2025) introduced a comprehensive nutritional advisory system using multimodal deep learning, which illustrates the extensive research efforts in this domain.

Previous work has been done to fine-tune a VLM to estimate calories for food images (Yao et al., 2024a). Similarly, "Snap-n-Eat," a system designed and built by Zhang et al. (2015), is a mobile food recognition system that recognizes food items on a plate to estimate portion sizes and calculate caloric and nutritional content. Combining these ideas, our ultimate unique goal will be food image analysis for bioavailable iron.

3 Data

MM-Food-100K comes from a larger dataset consisting of approximately 1.2M high-quality food photos with annotations that include ingredient portion sizes and nutrition information, which are commonly lacking in other food image datasets (Dong et al., 2025). We chose this dataset for its "real-world" food images that capture a wider breadth of information and cuisines. This includes many Asian dishes, for which nutrition is more commonly incorrectly estimated compared to other cuisines, according to Li et al. (2024).

Other popular food datasets, such as Recipe1M+, include photos of food scraped from online recipes (Marin et al., 2019), which differ greatly from more "realistic" photos taken by everyday people. Our research aimed to find food images that were very similar to or exactly those taken by people with their personal cameras or phones.

Food Category	% of Data
homemade food	46.56
restaurant food	35.46
raw vegetables and fruits	9.36
packaged food	8.35
others	0.27

Table 1: MM-Food-100K Food Category Breakdown

Unlike most other food datasets, MM-Food-100K also encapsulates many different food categories, as shown in Table 1. We took a smaller portion of it, which consisted of 5K images derived from stratified sampling based on food categories.

3.1 Data Acquisition and Augmentation

The original data included 11 columns, with 6 being relevant to our task:

`image_url` The source image URL.

`dish_name` The name of the dish pictured.

`food_type` The category of food pictured.

`portion_size` JSON of ingredient measurements: `{'fish': '300g', 'green onions': '20g'}`

`nutritional_profile` JSON of nutrition info:

```
{'fat_g': 20.0,
 'protein_g': 30.0,
 'calories_kcal': 350,
 'carbohydrate_g': 5.0}
```

`cooking_method` "steaming" or "Fried," etc.

The `nutritional_profile` column included calories and macronutrients, but was missing the micronutrients necessary to calculate bioavailable iron. To obtain iron, calcium and vitamin C data, we queried the USDA FoodData Central API, which is the "USDA's answer to the challenge of providing reliable, web-based, transparent, and easily accessible information about the nutrients and other components of foods" (Fukagawa et al., 2022).

3.2 Pre-Processing

The majority of pre-processing focused on acquiring and computing micronutrient measurements. First, we cleaned the data by lowercasing all strings throughout the relevant columns, including `portion_size`, which listed individual ingredients and their portions. From this column, we compiled a list of unique ingredient names and used them to obtain their respective iron, calcium, and vitamin C values per 100g serving from the USDA FoodData Central API. Then we added iron, calcium, and vitamin C columns to our data by calculating total amounts for each food image based on the portion sizes of their ingredients and the information retrieved per 100g of each ingredient.

We filled missing values, found only in the `cooking_method` column, with "unknown," and then split the data with a 80/10/10 ratio into train, validation, and test sets. Again, we used stratified sampling based on food categories to ensure balanced representation. This resulted in 4,000 samples for training and 1,000 samples each for validation and testing.

3.3 Question-Answer Pairs

Subsequently, we converted the data into Question-Answer (QA) pairs and stored them in JSON files, with some training examples presented in Listing 1. The result was 28,715 QA pairs for training, 3,601 for validation, and 3,586 for testing.

```
"question": "What is the general name of
    this dish?",
"answer": "pho",
"image_file": "train/7833510824700106693
    _961701_.jpeg"
...
"question": "What category of food is
    this?",
"answer": "restaurant food",
...
"question": "What primary cooking method
    was used for this dish?",
"answer": "boiled",
...
"question": "What is the estimated
    weight of the ingredient 'beef' in
    grams?",
"answer": 100.0,
...
"question": "Provide the numerical
    values for Calories (kcal), Fat (g),
    Protein (g), and Carbohydrates (g)
    for the entire dish, separated by
    commas.",
"answer": "450.0, 15.0, 30.0, 50.0",
...
"question": "Provide the numerical
    values for Total Weight (g), Iron (
    mg), Calcium (mg), and Vitamin C (mg
    ), separated by commas.",
"answer": "600.0, 3.0, 143.0, 7.0", ...
```

Listing 1: A snippet of the JSON training data

According to Chen et al. (2024), unintentional data leakage can occur during VLM training, since the questions could potentially contain or reveal answers. This would enable models to take shortcuts and even disregard information from visual content. With this in mind, we carefully crafted the questions and answers to ensure they are straightforward, requiring the model to depend on the provided image to respond.

As seen in the Listing 1, the questions were divided into six types. The first three asked about

dish name, food category, and cooking method. The remaining three pertained to portions of the ingredients, total weight of the ingredients, and estimates of the nutritional content based on the ingredient portions.

3.4 Exploratory Data Analysis

In this section, we present and analyze the distributions of key values within our training data. In addition to micronutrient measurements, we determined the total weight by summing the individual portion sizes of the ingredients for each dish. Table 2 provides the statistics for the total weight and micronutrients calculated for each image, and their distributions are shown in Figure 2. In addition, Table 3 shows statistics and Figure 3 shows the distributions for the calorie and macronutrient estimates provided in the original MM-Food-100K data.

Stats	Total Wt.	Iron	Calcium	Vit. C
Mean	366.89	4.00	142.55	20.13
Std.	180.39	4.23	126.99	30.74
Min	0.00	0.00	0.00	0.00
25%	250.00	1.71	46.00	0.75
50%	350.00	3.36	112.00	9.35
75%	450.00	5.31	202.00	25.01
Max	2050.00	152.50	1277.00	364.00

Table 2: Training Data Statistics for Total Weight (in g) and Micronutrients (in mg)

Stats	Calories	Fat	Protein	Carbs
Mean	412.67	17.58	21.30	40.86
Std.	258.17	14.80	18.22	29.83
Min	0.00	0.00	0.00	0.00
25%	250.00	8.00	5.00	20.00
50%	350.00	15.00	20.00	35.00
75%	600.00	25.00	30.00	60.00
Max	2500.00	180.00	200.00	400.00

Table 3: Training Data Statistics for Calories (in kcal) and Macronutrients (in g)

The micronutrient distributions are all extremely right-skewed, whereas the total weight distribution is more spread out. Calcium has a much higher mean and median than iron and vitamin C, while iron has the lowest mean and median overall. The calorie and protein distributions are more bimodal, while fat and carbs are more right-skewed, but not as much as the micronutrients. Carbs have the highest mean and median for macronutrients, while fat

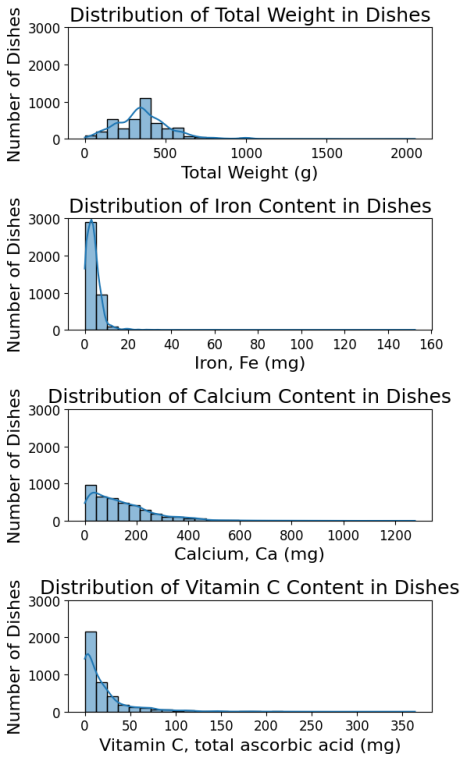

Figure 2: Training Data Distributions for Total Weight and Micronutrients

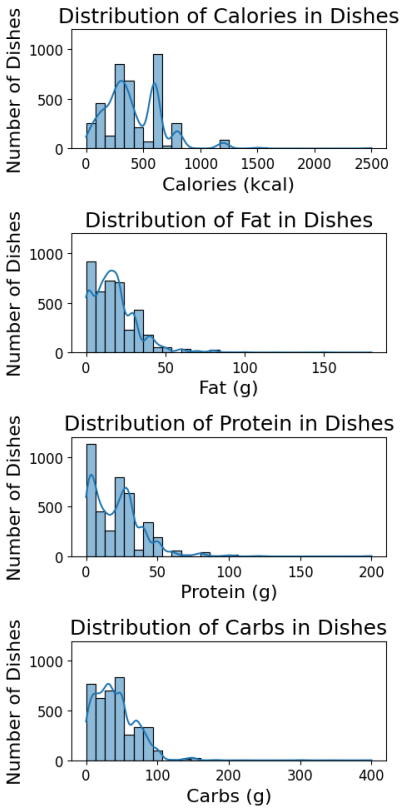

Figure 3: Training Data Distributions for Calories and Macronutrients

has the lowest of them, but only slightly lower than protein. Overall, these statistics and distributions show how much of a challenge it will be to accurately estimate these measurements. Although addressing this data skew is a critical step, the exploration of mitigating these issues will be left for future work, as the goal of this initial study is to first establish this task and data-to-model pipeline and baseline.

4 Methods

4.1 Models

Given the task's complexity and the limitations on time and resources, we opted to finetune existing pre-trained VLMs instead of building and training transformer models from scratch. Tian et al. (2025) conducted a thorough survey of advances and potential in LLMs for nutritional analysis and concluded that while hallucination still exists with very large models such as ChatGPT-4, further research in this field should prioritize "lightweight architectures." With this research, our aim was to explore the use of smaller models that could possibly fit and be used on mobile devices. We sought candidates around 2 to 3 billion parameters or less. These included PaliGemma (Beyer et al., 2024), Qwen2.5-VL (Bai et al., 2025), MiniCPM-V (Yao et al., 2024b), and SmolVLM (Marafioti et al., 2025) models and Table 4 shows the number of parameters per model.

Model	# Params
PaliGemma-3B	∼3B
Qwen2.5-VL-3B	∼3.8B
MiniCPM-V 2.0	∼3B
SmolVLM	256M/2.2B

Table 4: Model Comparison Summary

We chose to proceed with SmolVLM, which had the least parameters with two small sizes: 256M and 2.2B. SmolVLM models use SmolLM2, which is a small state-of-the-art language model (LM) with its largest size (1.7B parameters) outperforming other similarly small LMs such as Qwen2.5-1.5B and Llama3.2-1B (Allal et al., 2025).

4.2 Finetuning

In order to train our chosen models for our task, we employed LoRA (Low-Rank Adaptation) (Hu et al., 2021), a type of Parameter-Efficient Fine-Tuning (PEFT) (Xin et al., 2025) technique, which allowed smaller portions of our models to be trained in a very efficient manner. To do this, we used Pytorch, as well as the PEFT and Hugging Face transformers libraries. Our training objective was a standard cross-entropy loss over the VQA response sequence.

5 Experiments

We took advantage of free student units in Google Colab and used the A100 High-RAM GPU (80 GB), which allowed us to accelerate training and handle large data batches. Additionally, via Hugging Face, we were about to use bfloat16 with CUDA to save memory and computation time.

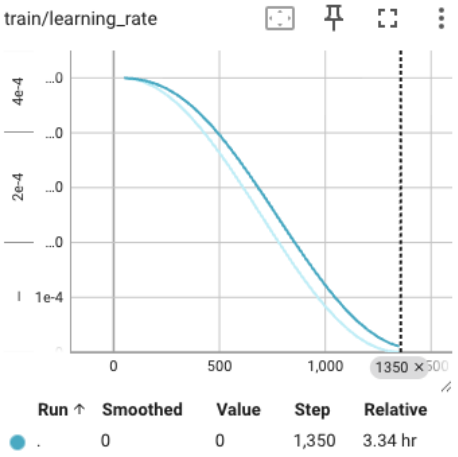

Figure 4: 2.2B SmolVLM-Instruct Learning Rate

5.1 SmolVLM-256M-Instruct

First, we fully set up our data processing, training, and validation pipelines and confirmed that they were working correctly with very small data subsets and the smaller SmolVLM-256M-Instruct model. Then, we performed the full finetuning of SmolVLM-256M-Instruct for 1 epoch, 128 batch size, 32 for both LoRA rank and alpha, and 0 for LoRA dropout. We used a learning rate scheduler with cosine decay that gradually decreased learning rate, as seen in Figure 4, with an initial rate of 0.0005 and a 0.3 warm-up ratio. This training took 34 minutes and 58 seconds and then benchmarking it on the validation set took 24 minutes and 38 seconds with a batch size of 256.

5.2 SmolVLM-Instruct (2.2B Parameters)

Next we trained the larger 2.2B SmolVLM-Instruct for 2 epochs, again with 128 batch size, 32 for both

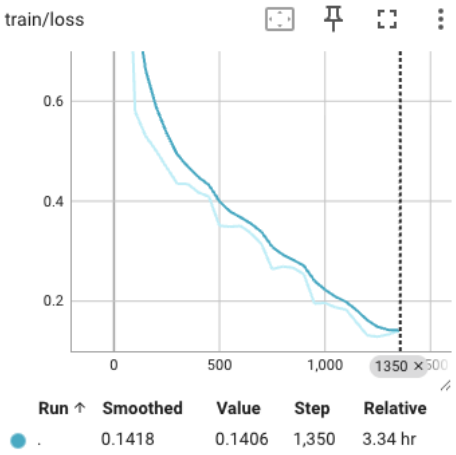

Figure 5: 2.2B SmolVLM-Instruct Training Loss

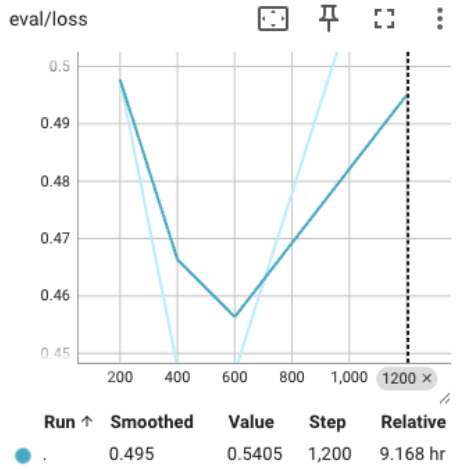

Figure 6: 2.2B SmolVLM-Instruct Validation Loss

LoRA rank and alpha, and 0 for LoRA dropout. As in the smaller model, we used the learning rate scheduler with cosine decay and the same initial rate of 0.0005 and 0.3 warm-up ratio. This training took 1 hour, 19 minutes, and 10 seconds. Since the training and evaluation loss curves did not show overfitting or underfitting, we extended training to 6 total epochs with all hyperparameters kept the same. Training 6 epochs took a total of 3 hours, 31 minutes, and 35 seconds. During training, the model was evaluated every 200 steps and the training and evaluation loss curves in Figures 5 and 6 show that the model started to overfit after step 600, somewhere in between the 2nd and 3rd epochs when the validation loss began to steeply rise. Due to this, we decided to do early stopping and saved the best model at step 600, which had the lowest validation loss, and then benchmarked it on the validation set, which took 24 minutes with the same validation batch size of 256.

6 Results and Discussion

6.1 Classification Results

The classification predictions were separated into three groups: Dish Name, Category (of food), and Cooking Method. In addition, the total accuracy was calculated from all three. Tables 5 and 6 show the classification results from validation for SmolVLM-256M-Instruct and 2.2B SmolVLM-Instruct models respectively.

	Baseline		Finetuned	
Accuracy	**Fuzzy**	**Strict**	**Fuzzy**	**Strict**
Dish Name	0.010	0.000	0.192	**0.182**
Category	0.000	0.000	0.794	0.794
Cook Method	0.004	0.000	0.440	0.440
Total	0.005	0.000	**0.476**	0.472

Table 5: SmolVLM-256M-Instruct Baseline and Finetuned Model Performance Comparison (Classification)

	Baseline		Finetuned	
Accuracy	**Fuzzy**	**Strict**	**Fuzzy**	**Strict**
Dish Name	0.116	0.000	0.354	**0.336**
Category	0.000	0.000	0.882	0.882
Cook Method	0.089	0.000	0.552	0.540
Total	0.068	0.000	**0.596**	0.586

Table 6: 2.2B SmolVLM-Instruct Baseline and Finetuned Model Performance Comparison (Classification)

As expected, the larger 2.2B SmolVLM-Instruct had higher finetuned accuracy across the board, achieving a 25.21% relative increase in finetuned fuzzy total accuracy compared to the smaller model. The most substantial relative improvement in comparing the finetuned models was observed in strict dish name accuracy, reaching 84.62%.

6.2 Regression Results

Regression predictions included total weight, micronutrient, calorie, and macronutrient values. Tables 7 and 8 show the regression results from validation for SmolVLM-256M-Instruct and 2.2B SmolVLM-Instruct models respectively. Once again, finetuning achieved significant results for both models. One surprising thing to note is that the larger 2.2B SmolVLM-Instruct baseline model performed significantly worse than the smaller baseline model. A hypothesis for this is possible variation in instruction tuning that caused differences in response generation. Despite this, 2.2B SmolVLM-

Instruct achieved lower finetuned MAE and RMSE overall.

	Baseline		Finetuned	
Regression	**MAE**	**RMSE**	**MAE**	**RMSE**
Total Wt.	354.32	391.67	103.37	180.44
Iron	7.44	15.69	2.42	3.79
Calcium	139.86	186.42	97.66	146.20
Vit. C	19.99	37.81	17.90	36.85
Calories	277.52	372.98	105.09	172.07
Fat	55.34	66.35	6.00	9.48
Protein	54.13	64.50	7.47	10.25
Carbs	54.57	62.66	16.63	25.44

Table 7: SmolVLM-256M-Instruct Baseline and Finetuned Model Performance Comparison (Regression)

	Baseline		Finetuned	
Regression	**MAE**	**RMSE**	**MAE**	**RMSE**
Total Wt.	316.50	370.56	72.40	127.38
Iron	537.38	713.72	1.54	2.91
Calcium	498.18	629.79	66.69	104.06
Vit. C	272.43	469.52	12.72	30.27
Calories	276.53	396.92	56.96	119.97
Fat	10.35	16.14	3.34	6.43
Protein	13.89	19.98	3.89	7.08
Carbs	192.08	375.70	8.85	17.95

Table 8: 2.2B SmolVLM-Instruct Baseline and Finetuned Model Performance Comparison (Regression)

6.3 Final Model Test Results

Our finetuned 2.2B SmolVLM-Instruct proved to be the best model in both classification and regression during validation. We proceeded to perform the final evaluation on this model with unseen data, our hold-out test set. Tables 9 and 10 show the classification and regression test set results for our finetuned 2.2B SmolVLM-Instruct.

Accuracy	**Fuzzy**	**Strict**
Dish Name	0.337	0.319
Category	0.846	0.846
Cook Method	0.576	0.572
Total	**0.587**	0.580

Table 9: 2.2B SmolVLM-Instruct Finetuned Model Test Data Performance (Classification)

As anticipated, the final model performed similarly well on the test set compared to the validation set. Specifically, the model achieved 0.587 total classification accuracy and MAE of 1.45 mg for iron. Although we observed improvements in metrics after finetuning, how can we be certain that our final model genuinely grasped the intended knowledge for the task? In an attempt to address this, we further evaluated our model by analyzing its responses compared to those of its baseline.

Regression	**MAE**	**RMSE**
Total Wt.	68.72	118.02
Iron	**1.45**	2.24
Calcium	67.06	120.59
Vit. C	12.47	27.74
Calories	64.10	148.29
Fat	4.04	8.94
Protein	4.48	9.74
Carbs	8.76	17.87

Table 10: 2.2B SmolVLM-Instruct Finetuned Model Test Data Performance (Regression)

First, we randomly sampled the test data and got the "noodle soup with fried chicken" pictured in Figure 7. Tables 11 to 16 show highlights of our QA results from the baseline and finetuned 2.2B SmolVLM-Instruct models.

Figure 7: An image of noodle soup with fried chicken

QA/Model	**Text**
Question	What is the general name of this dish?
Gold Answer	noodle soup with **fried chicken**
Baseline	Soup.
Finetuned	noodle soup with **egg and tomato**

Table 11

As seen in Table 11, the model proved to be better at capturing fine-grained details compared to the baseline, but it seemed to miss the fried chicken in the top right corner of the photo (Figure 7). However, in Table 12, the model accurately

estimated the size of the fried chicken portion when asked.

QA/Model	Text
Question	What is the estimated weight of the ingredient "fried chicken" in grams?
Gold Answer	150.0
Baseline	1000.
Finetuned	**150.0**

Table 12

QA/Model	Text
Question	What category of food is this?
Gold Answer	homemade food
Baseline	Soup.
Finetuned	**homemade food**

Table 13

QA/Model	Text
Question	What primary cooking method was used for this dish?
Gold Answer	boiling **and frying**
Baseline	Boiling.
Finetuned	boiling

Table 14

For the food category, the finetuned model got the exact answer of 150.0 (Table 13), but missed one detail for the cooking method, again related to the fried chicken it initially appeared to miss. Similarly to the baseline model, it answered "boiling" instead of "boiling and frying" (Table 14).

For calories and macronutrients (Table 15), the finetuned model got the exact answers while the baseline was off 50% or more for each value. For total weight and micronutrients (Table 16), the response of the finetuned model had mostly reasonable estimates with 3 out of 4 values within 10% of the answers, while the baseline model seemingly made wild guesses with values of 100 or 1000.

Next, we randomly sampled the test data once more and got the "crawfish boil" pictured in Figure 8. Tables 17 to 20 show the resulting QA responses that we found most interesting.

QA/Model	Text
Question	Provide the numerical values for Calories (kcal), Fat (g), Protein (g), and Carbohydrates (g) for the entire dish, separated by commas.
Gold Answer	600.0, 20.0, 25.0, 80.0
Baseline	300, 10, 10, 200.
Finetuned	**600.0, 20.0, 25.0, 80.0**

Table 15

QA/Model	Text
Question	Provide the numerical values for Total Weight (g), Iron (mg), Calcium (mg), and Vitamin C (mg), separated by commas.
Gold Answer	500.0, 5.3, 152.0, 12.95
Baseline	100, 1000, 1000, 100.
Finetuned	**450.0**, 3.9, **140.0, 12.75**

Table 16

Figure 8: An image of a crawfish boil

The finetuned model got the correct answer for the dish name while the baseline model responded with "Crab with garlic and ginger" (Table 17). However, when asked about the estimated weight in grams of a non-existing ingredient, "pizza," both models responded with values of 100.

Lastly, for calories and macronutrients (Table 19), the finetuned model again got the exact an-

QA/Model	Text
Question	What is the general name of this dish?
Gold Answer	crawfish boil
Baseline	The general name of this dish is "Crab with garlic and ginger."
Finetuned	**crawfish boil**

Table 17

QA/Model	Text
Question	What is the estimated weight of the ingredient "pizza" in grams?
Gold Answer	0.0
Baseline	**100.**
Finetuned	**100.0**

Table 18

QA/Model	Text
Question	Provide the numerical values for Calories (kcal), Fat (g), Protein (g), and Carbohydrates (g) for the entire dish, separated by commas.
Gold Answer	350.0, 10.0, 25.0, 30.0
Baseline	Calories: 1000, Fat: 100, Protein: 100, Car
Finetuned	**350.0, 10.0, 25.0, 30.0**

Table 19

QA/Model	Text
Question	Provide the numerical values for Total Weight (g), Iron (mg), Calcium (mg), and Vitamin C (mg), separated by commas.
Gold Answer	750.0, 3.26, 69.0, 30.65
Baseline	100, 1000, 1000, 100.
Finetuned	**400.0**, 1.6, 39.0, 12.75
Finetuned ∗ 1.875	***750.0***, ***3.0***, ***73.1***, *23.9*

Table 20: Test results for the "crawfish boil" image with a "Finetuned * 1.875" row added to apply a correction factor based on the total weight discrepancy

swers while the baseline model had the incorrect response format and got cut off while giving values of only either 1000 or 100. For total weight and micronutrients (Table 19), the baseline model again seemingly made wild guesses with values of 100 or 1000, while the finetuned models response was off by a lot. However, when multiplying the finetuned response values by 1.875 to correct the total weight, the iron and calcium measurements become within 10% of the answer values, showing that the micronutrient answers were off in proportion to the total weight. This implies that the model still demonstrates a strong capability for proportional estimation.

6.4 Future Work

As mentioned in our Exploratory Data Analysis, mitigating the data skew issues would be the ideal first step in future work. In the future, we would explore normalizing the numerical data and/or modifying the loss function with weighting to help improve performance in regression.

Other ideas for future work include exploring other methods such as few-shot learning (Parnami and Lee, 2022) using in-context learning (Dong et al., 2024), which involves providing examples of QA pairs within prompts for the model to learn from before responding. A great candidate for this would be a much larger VLM such as those of the Flamingo family. The Flamingo models incorporate a vision encoder that was pre-trained similarly to CLIP, which enabled powerful zero-shot learning (Radford et al., 2021), and led to the groundbreaking few-shot learning capabilities of Flamingo (Alayrac et al., 2022).

Lastly, ensuring balanced Visual Question Answering (VQA) datasets can help to resolve the issue of models taking shortcuts and relying on language biases instead of taking into account visual information (Goyal et al., 2017). This means before attempting to use more of the leftover MM-Food-100K data to train more models, we should investigate the balance of the dataset for our specific task. Originally, we used stratified sampling based on food category to take a subset and split the data. To have a more balanced dataset, we would need to ensure that each question is "associated with not just a single image, but rather a pair of similar images that result in two different answers to the question," just as Goyal et al. (2017) had done.

7 Conclusion

Understanding and accurately estimating bioavailable iron factors in food is crucial to solving iron deficiency. We used VLMs for food image analysis in relation to this task and demonstrated how to achieve a baseline solution through data engineering and finetuning with LoRA.

Our research made two key contributions to the area of food image analysis. First, we created a novel VQA dataset with QA pairs related to dish details, portion estimates, and nutrient measurements relevant to bioavailable iron calculation. We did this by retrieving iron, calcium, and vitamin C data using the USDA FoodData Central API for individual ingredients. We used the data to augment a subset of MM-Food-100K, a dataset with a variety of cuisines, including Asian dishes that have been proven to be more difficult to analyze. Second, we achieved significant improvements over baseline models in finetuning small VLMs for the VQA task. Our final finetuned model achieved a total classification accuracy of 0.587 on the test set and substantially reduced MAE and RMSE across regression metrics. We showed that even a small model can perform reasonably well for food image analysis.

Despite these achievements, our research had limitations mainly due to the highly skewed micronutrient distributions and the potentially unbalanced VQA data set, which make accurate prediction challenging. Future work will prioritize resolving these problems through techniques such as weighted loss functions and data normalization, as well as balancing the VQA data for more robustness and generalizability.

References

Abdulrahman Al-Naseem, Abdelrahman Sallam, Shamim Choudhury, and Jecko Thachil. 2021. Iron deficiency without anaemia: a diagnosis that matters. *Clinical Medicine*, 21(2):107–113.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L Menick, Sebastian Borgeaud, and 8 others. 2022. Flamingo: a visual language model for few-shot learning. In *Advances in Neural Information Processing Systems*, volume 35, pages 23716–23736. Curran Associates, Inc.

Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Martín Blázquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Hynek Kydlíček, Agustín Piqueres Lajarín, Vaibhav Srivastav, Joshua Lochner, Caleb Fahlgren, Xuan-Son Nguyen, Clémentine Fourrier, Ben Burtenshaw, Hugo Larcher, Haojun Zhao, Cyril Zakka, Mathieu Morlon, and 3 others. 2025. Smollm2: When smol goes big – data-centric training of a small language model. *Preprint*, arXiv:2502.02737.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, and 8 others. 2025. Qwen2.5-vl technical report. *Preprint*, arXiv:2502.13923.

Saikat Banerjee, Debasmita Palsani, and Abhoy Chand Mondal. 2024. Nutritional content detection using vision transformers- an intelligent approach. *International Journal of Innovative Research in Engineering and Management*, 11(6):21–27.

Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, Thomas Unterthiner, Daniel Keysers, Skanda Koppula, Fangyu Liu, Adam Grycner, Alexey Gritsenko, Neil Houlsby, Manoj Kumar, Keran Rong, and 16 others. 2024. Paligemma: A versatile 3b vlm for transfer. *Preprint*, arXiv:2407.07726.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, and Feng Zhao. 2024. Are we on the right way for evaluating large vision-language models? In *Advances in Neural Information Processing Systems*, volume 37, pages 27056–27087. Curran Associates, Inc.

Sheng-Tzong Cheng, Ya-Jin Lyu, and Ching Teng. 2025. Image-based nutritional advisory system: Employing multimodal deep learning for food classification and nutritional analysis. *Applied Sciences*, 15(9).

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Tianyu Liu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. 2024. A survey on in-context learning. *Preprint*, arXiv:2301.00234.

Yi Dong, Yusuke Muraoka, Scott Shi, and Yi Zhang. 2025. Mm-food-100k: A 100,000-sample multimodal food intelligence dataset with verifiable provenance. *Preprint*, arXiv:2508.10429.

Naomi K Fukagawa, Kyle McKillop, Pamela R Pehrsson, Alanna Moshfegh, James Harnly, and John Finley. 2022. Usda's fooddata central: what is it and why is it needed today? *The American Journal of Clinical Nutrition*, 115(3):619–624.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Leif Hallberg and Lena Hulthén. 2000. Prediction of dietary iron absorption: an algorithm for calculating absorption and bioavailability of dietary iron123. *The American Journal of Clinical Nutrition*, 71(5):1147–1160.

Ye He, Chang Xu, Nitin Khanna, Carol J. Boushey, and Edward J. Delp. 2013. Food image analysis: Segmentation, identification and weight estimation. In *2013 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6.

Ye He, Chang Xu, Nitin Khanna, Carol J. Boushey, and Edward J. Delp. 2014. Analysis of food images: Features and classification. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 2744–2748.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Richard Hurrell and Ines Egli. 2010. Iron bioavailability and dietary reference values1234. *The American Journal of Clinical Nutrition*, 91(5):1461S–1467S.

Landu Jiang, Bojia Qiu, Xue Liu, Chenxi Huang, and Kunhui Lin. 2020. Deepfood: Food image analysis and dietary assessment via deep model. *IEEE Access*, 8:47477–47489.

Xinyi Li, Annabelle Yin, Ha Young Choi, Virginia Chan, Margaret Allman-Farinelli, and Juliana Chen. 2024. Evaluating the quality and comparative validity of manual food logging and artificial intelligence-enabled food image recognition in apps for nutrition care. *Nutrients*, 16(15).

Andrés Marafioti, Orr Zohar, Miquel Farré, Merve Noyan, Elie Bakouch, Pedro Cuenca, Cyril Zakka, Loubna Ben Allal, Anton Lozhkov, Nouamane Tazi, Vaibhav Srivastav, Joshua Lochner, Hugo Larcher, Mathieu Morlon, Lewis Tunstall, Leandro von Werra, and Thomas Wolf. 2025. Smolvlm: Redefining small and efficient multimodal models. *Preprint*, arXiv:2504.05299.

Javier Marin, Aritro Biswas, Ferda Ofli, Nicholas Hynes, Amaia Salvador, Yusuf Aytar, Ingmar Weber, and Antonio Torralba. 2019. Recipe1m+: A dataset for learning cross-modal embeddings for cooking recipes and food images. *Preprint*, arXiv:1810.06553.

Chenrui Niu, Xiayang Ying, Gan Pei, Menghan Hu, and Guangtao Zhai. 2024. Review of the deep learning for food image processing. *International Journal of Agricultural and Biological Engineering*, 17(5):15–30.

Archit Parnami and Minwoo Lee. 2022. Learning from few examples: A summary of approaches to few-shot learning. *Preprint*, arXiv:2203.04291.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Sergio Romero-Tapiador, Ruben Tolosana, Blanca Lacruz-Pleguezuelos, Laura Judith Marcos-Zambrano, Guadalupe X. Bazán, Isabel Espinosa-Salinas, Julian Fierrez, Javier Ortega-Garcia, Enrique Carrillo de Santa Pau, and Aythami Morales. 2025. Are vision-language models ready for dietary assessment? exploring the next frontier in ai-powered food image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 430–439.

Qingfeng Tian, Boyuan Wang, and Shanquan Chen. 2025. Large language models in nutritional recognition: A comprehensive review of applications. In *2025 10th International Conference on Computer and Communication System (ICCCS)*, pages 78–82.

Yi Xin, Jianjiang Yang, Siqi Luo, Yuntao Du, Qi Qin, Kangrui Cen, Yangfan He, Zhiwei Zhang, Bin Fu, Xiaokang Yang, Guangtao Zhai, Ming-Hsuan Yang, and Xiaohong Liu. 2025. Parameter-efficient fine-tuning for pre-trained vision models: A survey and benchmark. *Preprint*, arXiv:2402.02242.

Dongyu Yao, Keling Yao, Junhong Zhou, and Yinghao Zhang. 2024a. Caloraify: Calorie estimation with visual-text pairing and lora-driven visual language models. *Preprint*, arXiv:2412.09936.

Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, Qianyu Chen, Huarong Zhou, Zhensheng Zou, Haoye Zhang, Shengding Hu, Zhi Zheng, Jie Zhou, Jie Cai, Xu Han, and 4 others. 2024b. Minicpm-v: A gpt-4v level mllm on your phone. *Preprint*, arXiv:2408.01800.

Weiyu Zhang, Qian Yu, Behjat Siddiquie, Ajay Divakaran, and Harpreet Sawhney. 2015. "snap-n-eat". *Journal of Diabetes Science and Technology*, 9(3):525–533.