Multimodal Learning for Calorie Estimation in Nutrition Study

Ajay Jagannath
Dept. of Computer Science
Texas A&M University
College Station, TX, USA
ajayjagan2511@tamu.edu

***Abstract*—Precise quantification of nutritional intake is a crucial challenge in understanding metabolic health and cardiovascular diseases prevention such as heart attacks. This paper aims to develop a novel multimodal deep learning architecture for lunch calorie estimation by integrating diverse data sources, including Continuous Glucose Monitoring (CGM) data, demographic information, microbiome profiles, meal images, and breakfast nutritional parameters. By combining these heterogeneous data modalities, the proposed model seeks to provide a more comprehensive and precise assessment of individual dietary habits. This enhanced calorie estimation is expected to contribute significantly to early detection and prevention strategies for cardiovascular diseases by identifying high-risk individuals based on their nutritional patterns. By leveraging advanced health analytics, this work highlights the transformative potential of integrating multiple data sources for improved nutritional analysis and heart attack prediction.**

I. Introduction

Cardiovascular diseases (CVDs), including heart attacks, represent the leading cause of mortality worldwide [1]. Diet plays a critical role in the development and progression of CVDs, with unhealthy eating habits emerging as a significant modifiable risk factor [2]. Accurate monitoring of dietary intake is essential for assessing cardiovascular risk and implementing preventive strategies. Traditional methods of dietary assessment, such as self-reported food diaries and recall questionnaires, consistently demonstrate unreliability due to systematic underreporting and inherent subjective biases [3]. These fundamental limitations underscore the urgent necessity for more objective and precise tools for dietary intake estimation, which can substantively aid in heart attack diagnosis and prediction.

Technological advancements have dramatically expanded our capacity to collect diverse data types related to dietary habits and metabolic health. Continuous Glucose Monitoring (CGM) systems now provide real-time, granular insights into an individual's glucose levels, offering a dynamic reflection of metabolic responses to food intake [4]. CGM data can indirectly shed light on critical information about caloric consumption, which is fundamental to cardiovascular health. Moreover, demographic factors and gut microbiome compositions have emerged as sophisticated indicators linked to dietary behaviors and metabolic health, suggesting their profound potential in enhancing predictive models for calorie estimation and cardiovascular risk assessment [5] [6].

Visual analysis of meal photographs has rapidly emerged as a promising frontier in dietary assessment. The advent of deep learning and advanced computer vision techniques has rendered image-based food recognition and calorie estimation increasingly feasible. Pretrained models like the Vision Transformer (ViT) have demonstrated exceptional performance in image classification tasks and can be strategically leveraged for feature extraction in dietary analysis [7]. By accurately identifying food items and portion sizes, visual data can significantly enhance the estimation of caloric and nutrient intake, a crucial component in assessing cardiovascular risk [8].

Transformers [9], originally conceptualized for natural language processing, have exhibited remarkable capabilities in capturing long-range dependencies and contextual information. Their inherent adaptability has facilitated successful applications in time-series analysis [10], rendering them particularly suitable for modeling CGM data. Similarly, transformers demonstrate proficiency in processing textual and categorical data, providing a unified architectural approach for handling multiple data modalities. Integrating these diverse data sources through multimodal learning can offer a comprehensive understanding of an individual's dietary patterns and metabolic responses.

Despite these technological advances, integrating heterogeneous data sources into a cohesive predictive model remains a significant computational and methodological challenge. Multimodal learning aims to exploit complementary information across various modalities to enhance model performance [11]. In the context of dietary assessment and cardiovascular health, combining CGM data, demographic and microbiome information, and meal images holds transformative potential for enhancing calorie estimation accuracy.

II. Methods

A. Data Description

There are several datasets in the project containing different multimodal data. These datasets are further divided into training and testing sets. The image data provides features such as images of food taken before breakfast/lunch and the amount of fiber consumed from both lunch and breakfast. Demographic information includes personal attributes of the subjects, such as age, gender, height, weight, BMI, Viome data, and other

facts relevant for calorie estimation. The Continuous Glucose Monitoring (CGM) dataset contains time series glucose data, and also include meal times of breakfast and lunch. The label dataset contains different feature columns, such as lunch calories, breakfast calories, lunch carbs, breakfast carbs, lunch protein, and breakfast protein. However, for the objective of this project, only lunch calories are considered.

B. Data Processing & Feature Selection

The initial challenge in advancing the project and predicting lunch calorie intake was ensuring the thorough preprocessing of the provided dataset. The demographic data was preprocessed by developing the function preprocess_demo, which splits the column Viome, a list of 26 floating-point values into different features, making them available to use downstream. Missing categorical values were replaced with the mode of the respective column, and the missing values in numerical columns were filled with the mean. Further, one-hot encoding combined with the categorical variables, and normalization of the numerical columns is done through MinMaxScaler. This function combines all the cleaned-up data into a single data frame.

For cleaning CGM sequence data, we created a function that would change the CGM Data column from strings into lists of floats; normalize all sequences in length by padding or truncating; and change Breakfast Time and Lunch Time to DateTime format. Further, missing times were replaced with column-wise mean, which were converted in seconds to increase compatibility. In the case of image processing, we created the function transform_images, which tries to give some standard structure to the image data. This function converts image strings to NumPy arrays, reshapes 2D arrays to 3D, and does several other transformations, for example, resizing and normalization. Also, it replaces invalid data with zero tensors to maintain dataset integrity. Similarly, we wrote the preprocess_img function, which, looking for anomalies across columns 'Image Before Breakfast' and 'Image Before Lunch', does transformations of individual images and fills N/A values in fiber columns with the rounded mean.

Next, to combine different datasets into one, we created the class MultiModalDataset implementation for regularizing the multimodal data. The class integrated CGM sequences, textual features, image data, and labels into one PyTorch-compatible structural space; to efficiently load and integrate multimodal data into machine learning workflows [12]. The merging of multiple datasets was done to create a comprehensive dataset. It sequentially combines CGM data, image data, and label data on "Subject ID" and "Day," and then merges the result with demographic data using Subject ID.

In developing our multimodal model, careful consideration was given to feature selection and model interpretability to ensure robust predictions. We removed features like subject_id and days to prevent overfitting, focusing on meaningful data that contributes to the model's generalization capabilities. Principal Component Analysis (PCA) was applied to the Viome data, reducing 26 features to 10 principal components. This dimensionality reduction retained critical variance while improving computational efficiency, allowing the model to focus on the most informative aspects of the data. For other features, we relied on respective transformer models to capture complex patterns and interactions. These models are adept at learning hierarchical representations, which enhance the understanding of each modality's contribution to the final predictions.

Finally, in feature assignment and labeling, we aligned the data to the different models, having structured the features and labels relevant to their particular requirements. For example, for image features, column features such as Image Before Breakfast and, Image Before Lunch were selected. Once features and labels were separated, the training and validation datasets were initialized using the MultiModalDataset class, ensuring compatibility with the multimodal data structure. This setup allowed seamless integration of CGM sequences, textual features, image data, and labels into the PyTorch data handling pipeline. The dataset was then efficiently loaded into PyTorch using the DataLoader utility, enabling batching, shuffling, and parallel data processing.[30] Through streamlined data preparation, we have optimized model training by adhering to PyTorch's standardized data processing workflows.[30]

C. Model Architecture

The multimodal model developed for this project integrates continuous glucose monitoring (CGM) sequences, demographic and textual features, and images to predict lunch calorie intake. This architecture leverages specialized neural network components tailored to learn the representation of each data modality, enabling effective feature extraction and integration. Then, we combine these representations into a joint embedding, which is used to make final predictions.

1) Time Series Transformer: For processing CGM sequences, we implemented a Time Series Transformer [13]. This component employs a Transformer architecture specifically designed for time-series data:

- Embedding Layer: A linear layer maps input CGM sequences to a higher-dimensional vector space suitable for processing by the Transformer.
- Positional Encoding: A tensor added to the input embeddings incorporates temporal information, crucial for capturing sequential dependencies.
- Transformer Encoder: Consisting multiple layers of multi-head self-attention and feed-forward networks, this encoder captures complex temporal patterns within the CGM data for feature extraction.
- Output Layer: A fully connected layer reduces the dimensionality of the transformed features, producing a compact representation of the CGM data.

2) Text Transformer: Demographic and textual features are processed using a Text Transformer, which also utilizes a Transformer architecture:

- Embedding Layer: Converts input textual features into an intermediate vector space representation.

- Positional Encoding: Similar to the Time Series Transformer, this adds positional context to the embeddings.
- Transformer Block: Processes the embedded features through layers of multi-head self-attention layers and feed-forward networks to learn the representation of the data.
- Mean Pooling: Aggregates sequence-level features into a fixed-size vector.
- Output Layer: Maps the pooled representation to the output space

3) Vision Transformer (ViT) Transformer: For image data, we utilize a ViT Encoder, based on Vision Transformer architecture:

- Pretrained Backbone: The ViT model is initialized with pretrained weights from 'Food101' Dataset [14] to leverage learned visual representations of various food categories.
- Backbone Freezing: The hidden layers can be frozen for fine-tuning with training data.
- Feature Extraction: The ViT processes images taken before breakfast and lunch separately, extracting high-level features from each.
- Sequential Classifier Head: Fully connected layer tailored to output image embeddings of a specified dimension.

4) Multimodal Integration: The final model combines the learned representations from all modalities into a single prediction pipeline:

- Feature Concatenation: Outputs from the Time Series Transformer, Text Transformer, and ViT Encoder are concatenated into a unified feature vector.
- Fully Connected Layers: This combined feature vector is processed through sequential linear layers with ReLU activations, cumulating into a single scalar output representing predicted lunch calories.

This architecture effectively encodes diverse data types by employing specialized components for each modality, comprehensively enhancing feature extraction and integration. The use of advanced architectures like Transformers and Vision Transformers ensures that complex patterns within each modality are captured, with multi-head self-attention capturing the contextual information. The joint vector space is decoded and utilized efficiently in the prediction task.

D. Training Procedure

The training procedure for our multimodal model involved experimenting with various hyperparameters to optimize performance and ensure robust learning across different data modalities. The key components of our training strategy are outlined below.

1) Hyperparameter Tuning: To balance memory usage and convergence speed, we evaluated mini-batch sizes of 16, 32, and 64. This exploration allowed us to identify the optimal batch size that facilitates efficient training without exhausting computational resources.

We also experimented with dropout probabilities ranging from 0.1 to 0.5 to mitigate overfitting. Dropout helps in preventing the model from becoming overly reliant on specific features, thus enhancing generalization.

For the Transformer components, we varied the number of hidden layers between 2 and 4. This adjustment aimed to capture complex non-linear patterns in the data by increasing the model's capacity to learn hierarchical representations.

2) Alternative Architectures: In addition to the primary model architecture, we explored alternative neural network configurations to assess their impact on performance:

- ResNet [15]: We replaced the Vision Transformer (ViT) with a ResNet architecture to compare their effectiveness in processing image data. ResNet's residual connections offer a different approach to feature extraction compared to ViT's attention mechanism. Owing to the small size of dataset, we found ResNet to perform better than an untrained Vit. However, pretrained Vit performed better even on the small dataset.
- Random Forest Ensemble [16]: By converting CGM data into statistical features such as mean, median, and standard deviation, we experimented with a random forest ensemble as an alternative to neural networks for decoding. This approach provided insights into the utility of traditional machine learning methods in handling specific data modalities.

Throughout training, we monitored performance metrics on both training and validation datasets to track progress and adjust hyperparameters as necessary. This iterative process ensured that our model was not only accurate but also generalizable across unseen data.

The final model configuration was selected based on its ability to achieve competitive performance on the validation set while maintaining computational efficiency.

III. Experiments and Results

A. Experimental Setup

The experimental setup for training and evaluating our multimodal model was designed to ensure robust performance and minimize bias. The experiments were conducted using a T4 GPU provided by Google Colab, which facilitated efficient computation and accelerated model training.

1) Training Configuration: The model was trained over a total of 30 epochs. We observed that convergence was typically achieved before reaching the maximum number of epochs. The use of a T4 GPU allowed for faster processing times, enabling extensive experimentation with different hyperparameters and model configurations. The model was trained using the AdamW optimizer, which combines adaptive learning rate capabilities with weight decay regularization to prevent overfitting. The learning rate was fine-tuned through experimentation to ensure stable convergence.

The Root Mean Square Relative Error (RMSRE) was employed as the loss function, defined as:

$$RMSRE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\frac{\hat{y}_i - y_i}{y_i + c}\right)^2}$$

where $\hat{y}_i$ represents the predicted value, y_i is the true value and c is small constant value used for latching. RMSRE is particularly suited for this task as it emphasizes relative errors, which are scale-invariant given the varying magnitudes of target values.

2) Validation Strategy: To validate the model's performance and ensure generalization across different subjects, we employed a holdout validation strategy. Specifically, we held out all data from several subjects chosen at random as the validation set. This approach prevented any potential leakage of information from the training set to the validation set, thereby minimizing bias in performance evaluation.

By holding out entire days of data from specific subjects, we ensured that the validation set remained independent from the training data. This strategy provided a more realistic assessment of the model's ability to generalize to new, unseen data.

B. Results

The performance of our multimodal model was evaluated using the Root Mean Square Relative Error (RMSRE) loss over 30 epochs. Figure 1 illustrates the training and validation RMSRE across epochs, demonstrating the model's learning progression and stability.

- Training RMSRE: The model's training RMSRE decreased consistently from an initial value of 0.9910 to 0.3292 by the final epoch.
- Validation RMSRE: Similarly, the validation RMSRE improved from 0.9740 in the first epoch to 0.3196 in the last epoch, indicating effective generalization to unseen data.

The testing phase yielded an RMSRE of 0.3274, confirming that the model retained its performance on a separate test set.

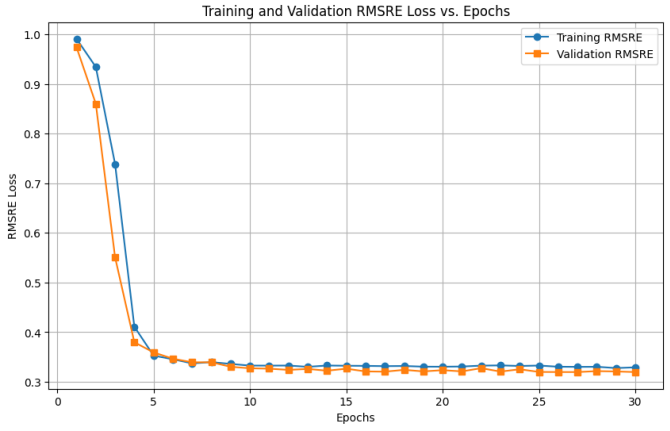

Fig. 1. Training and Validation RMSRE per Epoch. The graph illustrates the convergence of the model over 30 epochs, showing a steady decrease in both training and validation RMSRE.

C. Comparison with Baseline

To assess the effectiveness of our multimodal approach, we compared its performance against a baseline model. The baseline model achieved an RMSRE of 0.5258 on the test set.

Our multimodal model outperformed this baseline by 37.73%, achieving a test RMSRE of 0.3274. Integrating multiple data modalities through advanced neural network architectures, helps achieve greater accuracy in calorie prediction by leveraging diverse feature representations.

The comparison underscores the efficacy of our approach in capturing complex patterns and relationships within the data, which are not readily accessible through simpler models or single-modality inputs.

IV. Conclusion

We successfully developed a multimodal model for predicting lunch calorie intake by integrating continuous glucose monitoring (CGM) sequences, demographic and textual features, and images. Utilizing advanced neural network architectures, such as Transformers and Vision Transformers, the model effectively captured complex patterns across diverse data modalities. The preprocessing techniques ensured high-quality data input, while the custom MultiModalDataset class facilitated seamless integration of multimodal data.

The model achieved a significant improvement over baseline performance, with a test RMSRE of 0.3274 compared to the baseline's 0.5258, marking a 37.73% enhancement. This underlines the model's ability to generalize well across different subjects and data types. Hyperparameter tuning and experimentation with alternative architectures, including ResNet and random forest ensembles, provided valuable insights into optimizing model performance.

V. Future Scope

Building upon successfully predicting lunch calorie intake, future work will focus on expanding the model's capabilities to predict a broader range of nutritional macros, including carbohydrates, fats, and proteins. This extension aims to provide a more comprehensive nutritional analysis, which is crucial for detailed dietary assessments and informed decision-making regarding nutrition and health.

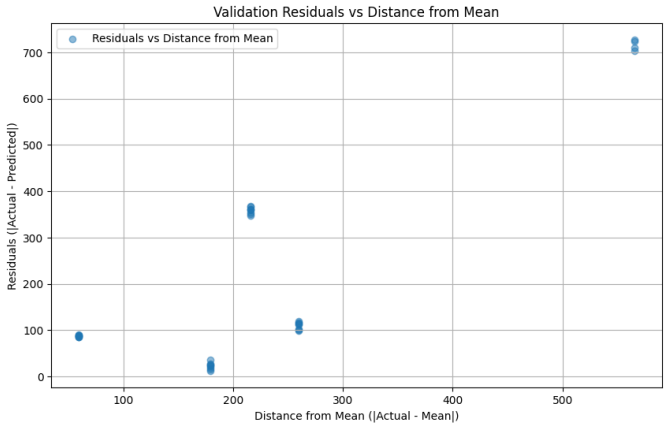

Fig. 2. Validation Residuals vs Distance from Mean. The graph illustrates that residuals increase with greater distance from the mean, indicating less accurate predictions for extreme values.

A. Bias Towards Mean in Predictions

Upon analyzing the validation predictions, we observed a tendency for predicted values to gravitate towards the mean of the dataset. This behavior results in less accurate predictions for values that deviate significantly from the mean, as shown in Figure 2.

This observation suggests that the model may be biased towards central tendency, potentially due to an imbalance in data distribution or limitations in capturing complex patterns for outliers. Addressing this issue is crucial for improving model accuracy, particularly for extreme values.

Future improvements could involve augmenting the dataset to ensure a more balanced representation across different value ranges or enhancing model complexity to better capture variability. These steps would help reduce bias towards mean values and improve prediction accuracy across the entire range of possible outcomes.

REFERENCES

[1] WHO WHO. cardiovascular diseases (cvds). *World Health Organization (WHO)*, 2017.

[2] Lawrence J Appel, Jeanne M Clark, Hsin-Chieh Yeh, Nae-Yuh Wang, Janelle W Coughlin, Gail Daumit, Edgar R Miller III, Arlene Dalcin, Gerald J Jerome, Steven Geller, et al. Comparative effectiveness of weight-loss interventions in clinical practice. *New England Journal of Medicine*, 365(21):1959–1968, 2011.

[3] Michele N Ravelli and Dale A Schoeller. Traditional self-reported dietary instruments are prone to inaccuracies and new approaches are needed. *Frontiers in nutrition*, 7:90, 2020.

[4] María González-Rodríguez, Marcos Pazos-Couselo, José M García-López, Santiago Rodríguez-Segade, Javier Rodríguez-García, Carmen Túñez-Bastida, and Francisco Gude. Postprandial glycemic response in a non-diabetic adult population: the effect of nutrients is different between men and women. *Nutrition & metabolism*, 16:1–9, 2019.

[5] Jose M Ordovas, Lynnette R Ferguson, E Shyong Tai, and John C Mathers. Personalised nutrition and health. *Bmj*, 361, 2018.

[6] Rebecca L Walker, Hera Vlamakis, Jonathan Wei Jie Lee, Luke A Besse, Vanessa Xanthakis, Ramachandran S Vasan, Stanley Y Shaw, and Ramnik J Xavier. Population study of the gut microbiome: associations with diet, lifestyle, and cardiometabolic disease. *Genome medicine*, 13:1–16, 2021.

[7] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[8] Quin Thames, Arjun Karpur, Wade Norris, Fangting Xia, Liviu Panait, Tobias Weyand, and Jack Sim. Nutrition5k: Towards automatic nutritional understanding of generic food. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8903–8911, 2021.

[9] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

[10] Bryan Lim and Stefan Zohren. Time-series forecasting with deep learning: a survey. *Philosophical Transactions of the Royal Society A*, 379(2194):20200209, 2021.

[11] Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. Multimodal machine learning: A survey and taxonomy. *IEEE transactions on pattern analysis and machine intelligence*, 41(2):423–443, 2018.

[12] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[13] Qingsong Wen, Tian Zhou, Chaoli Zhang, Weiqi Chen, Ziqing Ma, Junchi Yan, and Liang Sun. Transformers in time series: A survey. *arXiv preprint arXiv:2202.07125*, 2022.

[14] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part VI 13*, pages 446–461. Springer, 2014.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[16] Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.