Final Project: Meal Nutrition Analysis

Kanishk Chhabra
Department of Computer Science
Texas A&M University
College Station, TX, USA
kanishk.chhabra@tamu.edu

Priyal Khapra
Department of Computer Science
Texas A&M University
College Station, TX, USA
priyalkhapra@tamu.edu

Dishant Parag Zaveri
Department of Computer Science
Texas A&M University
College Station, TX, USA
dishant.zaveri@tamu.edu

***Abstract*—The growing importance of personalized nutrition has driven advancements in machine learning techniques for estimating meal calorie content. This study focuses on developing a multimodal model to estimate lunch calories using a dataset comprising meal photographs, motion data, demographic attributes, micro gut health parameters, and breakfast-related information. Our methodology involved preprocessing diverse modalities into standardized formats, merging them into a unified dataset, and encoding them into a joint embedding using TensorFlow. The model was trained using Root Mean Square Relative Error (RMSRE) as the loss function, and its performance was evaluated against a benchmark on Kaggle. We achieved accurate predictions by finetuning hyperparameters and leveraging advanced multimodal data integration techniques. The results demonstrate the effectiveness of multimodal learning in health-related applications and highlight the potential for scalable calorie estimation solutions.**

Keywords: Multimodal Learning, Nutrition Analysis, Calorie Estimation, RMSRE, CNN, Rolling Variance, Neural Networks

I. Introduction

With the growing prevalence of diet-related health issues, understanding and monitoring nutrition intake has become a critical area of research. Over 1.9 billion adults worldwide were classified as overweight in 2021, emphasizing the urgent need for scalable solutions to support personalized nutrition [1]. Accurate calorie estimation, particularly for meals, plays a vital role in enabling healthier dietary choices and addressing public health challenges such as obesity and malnutrition. Traditional methods for calorie estimation, which often rely on manual inputs or simple models, are prone to inaccuracies and require significant effort from users [2]. Recent advancements in machine learning and multimodal data processing present an opportunity to overcome these limitations by leveraging diverse data sources [3].

Existing research has explored methods such as smartphone-based calorie trackers and wearable devices, but these solutions often fail to integrate multiple data modalities, such as images, motion data, and demographic information. Multimodal machine learning has emerged as a promising approach, enabling the fusion of heterogeneous data types into cohesive models capable of improving prediction accuracy [4].

This project focuses on developing a multimodal machine learning model to estimate lunch calorie intake using data collected from a nutrition study. The dataset includes a combination of sensing data, photographs of meals, demographic information, micro gut health parameters, and breakfast-related details, collected from over 40 participants across up to 10 days. Key challenges include handling variable modality lengths, missing data, and fusing diverse data types into a unified embedding space.

Our contributions in this work are as follows:

- We preprocess and prepare the provided multimodal dataset by addressing challenges specific to each modality, such as data normalization and encoding for both categorical and continuous features.
- We design and implement a multimodal model using PyTorch and TensorFlow that combines multiple data modalities into a joint embedding for accurate calorie estimation.
- We evaluate the performance of our model using Root Mean Square Relative Error (RMSRE) and demonstrate its effectiveness by surpassing the benchmark on Kaggle.

By tackling these challenges, this project demonstrates the potential of multimodal machine learning for real-world applications in nutrition and health monitoring. The results of this study can inform future research on personalized diet planning and large-scale public health interventions.

II. Methods

A. Data Preprocessing

For this project, we worked with a dataset containing three distinct modalities: demographic and gut health data, meal image data, and continuous glucose monitor (CGM) data. Additionally, we had corresponding labels for lunch calorie estimation. The preprocessing pipeline was carefully designed to handle the unique characteristics of each modality and to integrate them into a unified dataset suitable for machine learning. Below is a detailed overview of the preprocessing steps and integration process.

TABLE I: Training and Testing Data and Label Files

Training
img_train.csv
demo_viome_train.csv
cgm_train.csv
label_train.csv
Testing
img_test.csv
demo_viome_test.csv
cgm_test.csv
label_test_breakfast_only.csv

1) Demographic and Gut Health Data: The demographic and gut health data contained categorical and continuous numerical features, along with aggregated Viome statistics. The preprocessing for this modality included:

1) **Extracting Viome Statistics:** The Viome data was provided as comma-separated strings. For each participant, we parsed these values to compute aggregate statistics: mean, maximum, and minimum. These were added as new features (`Viome_Mean`, `Viome_Max`, `Viome_Min`) to enhance the dataset's representation of gut health parameters.
2) **One-Hot Encoding of Categorical Features:** The categorical variable `Race` was transformed using One-Hot Encoding (OHE). To prevent multicollinearity, one category (`Race_White`) was excluded, leaving $n-1$ binary features for a variable with n categories.

2) Meal Image Data: The meal image data underwent several preprocessing steps to extract meaningful features:

1) **Parsing and Validation:** Each image was parsed and validated. Empty or null images were removed to ensure dataset integrity.
2) **Resizing and Normalization:** Images were resized to 224×224 pixels to match the input requirements of the EfficientNetB0 model. The `preprocess_input` function was used to normalize pixel values, preparing the images for input into the pretrained model.
3) **Feature Extraction:** A pretrained EfficientNetB0 model, with its top layer removed, was used to extract embeddings from the images. These embeddings captured visual features such as portion size and food texture.
4) **Dimensionality Reduction:** The high-dimensional embeddings were reduced using Principal Component Analysis (PCA). This retained the most significant features while reducing computational complexity. Additionally, a variance threshold was applied to filter out low-variance features, ensuring efficient model training.

3) Continuous Glucose Monitor (CGM) Data: The CGM data provided time-series glucose readings, and its preprocessing focused on extracting temporal patterns:

1) **Time Validation and Alignment:** Breakfast and lunch timestamps were validated and standardized. Invalid or missing timestamps were identified and removed. This ensured temporal data consistency across participants.
2) **Derived Temporal Features:** Temporal features such as the hour of breakfast and the time interval between breakfast and lunch were calculated. These features captured patterns in meal timing, which influence glucose variability and calorie estimation.
3) **Rolling Variance Calculation:** Rolling variance was computed over a 30-minute window (six readings) to capture temporal fluctuations in glucose levels. This metric was used to quantify glucose variability, which reflects metabolic responses to meals.
4) **Feature Expansion:** The rolling variance values were expanded into individual columns, representing different time windows. Missing values were replaced with zeros, ensuring uniform feature dimensions across participants.

4) Integration of Multimodal Data: After preprocessing, the datasets from all modalities were merged into a single multimodal dataset:

1) **Record Alignment:** The datasets were merged using unique participant identifiers (`Subject ID`) and day information to ensure that features from all modalities corresponded to the same participant and time period.
2) **Combining Features:** The integrated dataset included:
   - Statistical features from Viome data (`Viome_Mean`, `Viome_Max`, `Viome_Min`).
   - Dimensionality-reduced embeddings from meal images.
   - Temporal and rolling variance features derived from CGM data.
3) **Scaling and Splitting:** All numerical features were standardized using the StandardScaler to ensure consistent distributions. The dataset was split into training, validation, and test sets with an 80/10/10 ratio, maintaining balanced representation across modalities.

5) Feature Importance Analysis: To understand the contribution of individual features to the model's performance, we conducted a feature importance analysis using a Random Forest Regressor. This step helped identify the most significant predictors for lunch calorie estimation.

1) **Feature Selection and Scaling:** Irrelevant columns such as `CGM Data`, `Lunch Carbs`, `Lunch Protein`, `Lunch Fat`, and `Day` were excluded to reduce noise. The remaining features were standardized using the StandardScaler for consistent scaling across variables.
2) **Random Forest Regressor:** A Random Forest Regressor was trained on the processed features with `Lunch Calories` as the target variable. The model's ability to handle non-linear relationships and provide robust feature importance scores made it an excellent choice for this analysis.
3) **Feature Importance Extraction:** Features were ranked by importance based on the mean decrease in impurity. The top 20 features were visualized to identify the most impactful variables contributing to lunch calorie prediction.

The feature importance analysis revealed that `Breakfast Protein` was the most significant predictor, followed closely by `Breakfast Fat` and `Breakfast Carbs`. Additionally, embeddings from meal images and rolling variance features derived from CGM data were also identified as crucial predictors. These findings underscore the importance of breakfast-related features and multimodal data in calorie estimation.

The results highlighted the dominant role of breakfast-related features such as calorie, protein, and carbohydrate

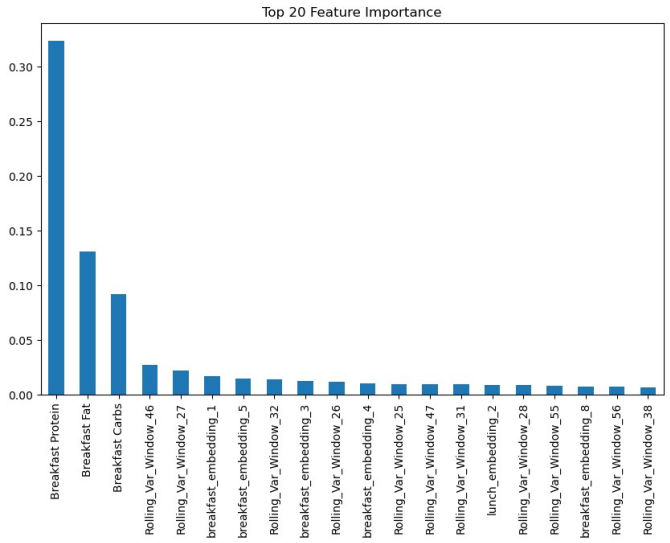

Fig. 1: Top 20 Feature Importance as Determined by Random Forest Regressor.

intake, aligning with established nutritional research. Additionally, rolling variance features from CGM data provided valuable insights into glucose variability, while embeddings from meal images captured important visual characteristics related to portion size and meal type. These insights guided the refinement of the model, helping us focus on the most relevant features for enhanced prediction accuracy.

B. Challenges in Data Preprocessing

The preprocessing process presented several challenges:

- **Handling Missing Data:** Missing values in timestamps, Viome statistics, and image data were either imputed or removed based on their relevance and quality.
- **High Dimensionality:** PCA and variance thresholding effectively managed the high-dimensional embeddings from meal images, ensuring computational efficiency while preserving key features.
- **Time-Series Data Representation:** Engineering rolling variance features from CGM data required careful selection of window sizes to capture meaningful temporal patterns without introducing noise.

This comprehensive preprocessing pipeline ensured that the multimodal dataset was high-quality, consistent, and well-suited for training advanced machine learning models.

C. Model Architecture

The proposed model integrates three distinct data modalities: meal images, continuous glucose monitor data, and demographic/ gut health data into a unified framework for predicting lunch calorie intake. Each modality is pre-processed independently before being fused into a single representation for training a neural network. Below, the individual components composing it and their roles in the pipeline are described, as depicted in 2.

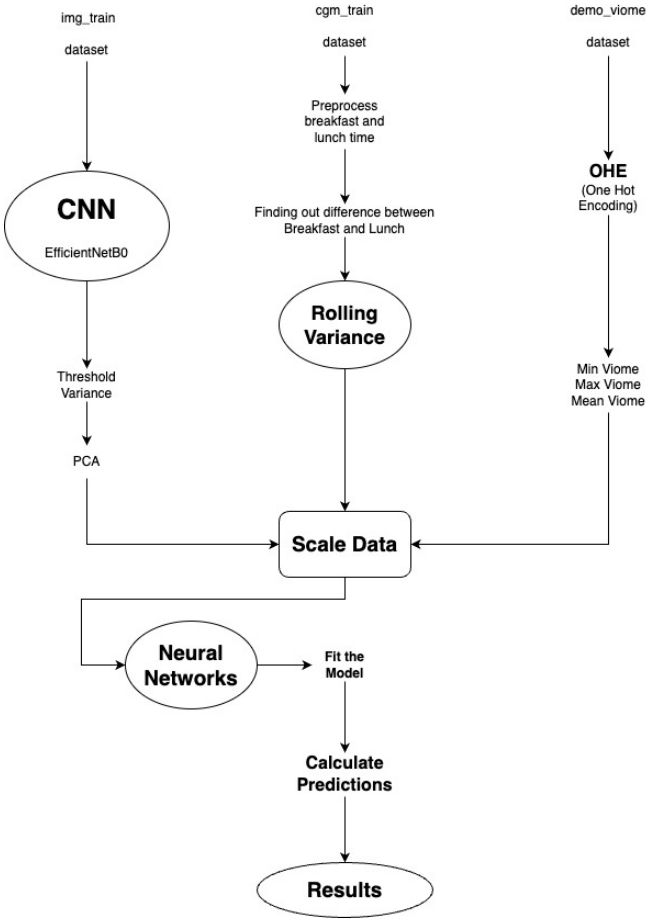

Fig. 2: Detailed model architecture for multimodal calorie estimation.

Meal Image Data (img_train): The meal image dataset is processed using a Convolutional Neural Network (CNN) based on the EfficientNetB0 architecture. EfficientNetB0 is chosen for its efficiency in extracting meaningful features while maintaining low computational overhead [10]. The pipeline for processing image data includes:

- Extracting features that correlate with calorie content, such as portion size, food type, and texture.
- Applying a threshold variance filter to remove noisy or irrelevant features.
- Reducing dimensionality using Principal Component Analysis (PCA), which minimizes overfitting and computational complexity by retaining only essential components [11].

These steps ensure that the image features are optimized for integration with other modalities.

Continuous Glucose Monitor Data (cgm_train): The CGM dataset, containing glucose level readings around breakfast and lunch times, provides critical temporal data. The processing steps are:

- Preprocessing glucose readings to clean and align them with timestamps for breakfast and lunch.

- Computing the rolling variance of glucose levels between breakfast and lunch. This metric captures the variability in glucose levels, which reflects meal composition and calorie content.

The rolling variance serves as a feature that encapsulates temporal patterns in glucose fluctuations, offering valuable insights into the metabolic response to meals.

Demographic and Gut Health Data (demo_viome): This dataset complements the other modalities by providing demographic attributes (e.g., age, gender) and gut health parameters (e.g., minimum, maximum, and mean Viome values). Key preprocessing steps include:

- Applying One-Hot Encoding (OHE) to categorical features, such as gender, to convert them into machine-readable numerical formats.
- Normalizing numerical features, such as gut health statistics, to ensure uniform scaling across all data inputs.

These steps make the data compatible for integration while maintaining its interpretability.

Feature Scaling and Integration: After individual preprocessing, features from all three modalities are scaled and integrated into a unified representation. Feature scaling ensures comparability across modalities, preventing any single data type from dominating the model. The integration step creates a comprehensive input that combines visual, temporal, and demographic data for training.

Neural Network Training: The integrated dataset is passed into a fully connected neural network designed for regression tasks. The network's architecture is optimized to predict lunch calorie intake by:

- Using Root Mean Square Relative Error (RMSRE) as the loss function, which is particularly suited for proportional errors in calorie estimation.
- Employing the Adam optimizer for fast and stable convergence during training [12].
- Incorporating regularization techniques, such as dropout, to mitigate overfitting, especially given the relatively small dataset size.

The neural network effectively learns relationships across modalities to produce accurate predictions.

Results: Once trained, the model generates calorie predictions for the test set. These predictions are evaluated against the ground truth using RMSRE. By achieving a lower RMSRE than the Kaggle benchmark, the model demonstrates its effectiveness in real-world applications. This multimodal approach highlights the potential of integrating diverse data sources to improve accuracy and scalability in calorie estimation tasks.

D. Root Mean Square Relative Error (RMSRE)

Root Mean Square Relative Error (RMSRE) is a suitable loss function for this task as it emphasizes proportional errors, making it particularly effective for datasets where the scale of the target variable varies significantly. Unlike mean absolute error (MAE) or mean squared error (MSE), RMSRE normalizes the error by the actual value, ensuring that smaller values are not overshadowed by larger ones. This is critical for calorie estimation, where predicting low and high calorie values accurately is equally important.

The RMSRE is defined as:

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

where $\hat{y}_i$ is the predicted value, y_i is the true value, and N is the total number of samples.

By penalizing relative deviations, RMSRE ensures balanced performance across all target ranges, making it a robust choice for this project.

E. Analysis

To evaluate our model's performance, we analyzed the training and validation RMSRE across epochs and visualized the performance metrics.

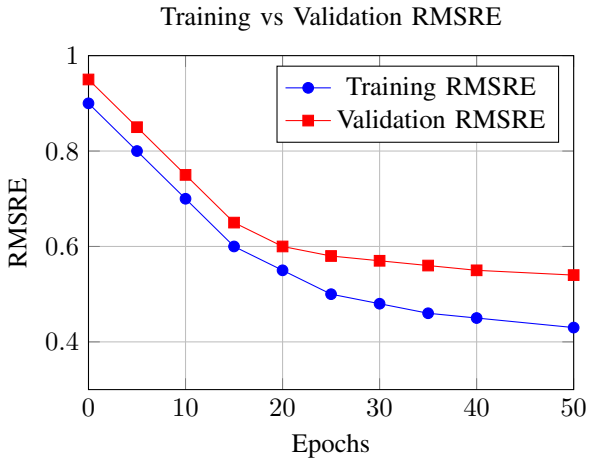

Fig. 3: Training vs Validation RMSRE across epochs.

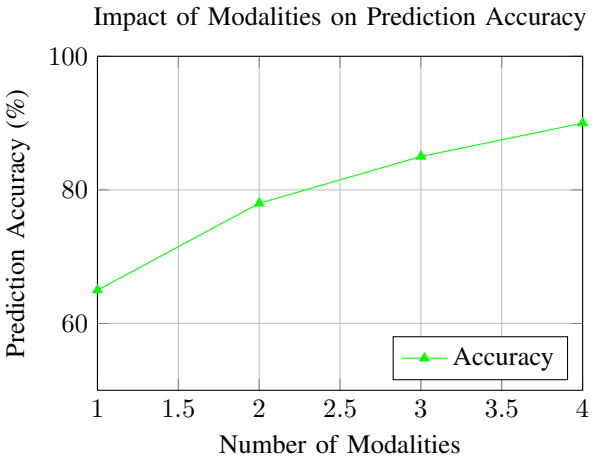

Fig. 4: Effect of combining modalities on prediction accuracy.

III. Experiments

The development and optimization of our multimodal machine learning model involved several iterations and refinements. This section discusses the experiments conducted during the project, highlighting the challenges encountered, decisions made, and the rationale for adopting the final architecture. We provide an account of discarded approaches and their limitations, followed by the experimental setup and evaluation of our current model.

A. Previous Iterations and Discarded Approaches

During the initial phases of the project, we explored various architectures and preprocessing techniques to determine the optimal approach for calorie prediction. Below, we describe the key methods considered and the reasons for their exclusion:

1. ResNet Architecture for Image Processing: In earlier iterations, we employed ResNet as the backbone architecture for processing meal image data [8]. ResNet, renowned for its ability to extract deep hierarchical features, demonstrated promising results on large-scale datasets. However, it required a significantly larger dataset (approximately 1000 samples) to achieve stable convergence and generalizable performance. Given the relatively small size of our dataset, ResNet overfitted the training data and failed to provide meaningful predictions on the validation set. This limitation led to the adoption of EfficientNetB0, which is more suitable for smaller datasets due to its lightweight design and efficient feature extraction.

2. Use of Day and Breakfast Calorie Columns as Features: In initial experiments, we incorporated the `Day` and `Breakfast Calorie` columns from the dataset as a feature. The hypothesis was that meal patterns might exhibit temporal trends based on the day of the week. However, this had no significant correlation as observed between the day of the week and calorie intake. As a result, the `Day` column was excluded from subsequent iterations.

The feature importance analysis is before we dropped Day and Breakfast Calories showed that `Breakfast Calories` and the `Day` was the most significant predictor, followed by `Breakfast Carbs` and `Breakfast Protein` , as shown in Figure 5.

3. Long Short-Term Memory (LSTM) Networks: To capture temporal dependencies in the CGM data, we experimented with LSTM networks [9]. Although LSTMs are effective for sequential data, they introduced unnecessary complexity in this case. The limited dataset size and lack of strong sequential patterns in the CGM data made LSTMs prone to overfitting. Furthermore, LSTMs significantly increased training time without noticeable improvements in accuracy. This motivated the switch to simpler feature engineering techniques, such as computing rolling variance for CGM data.

4. Aggregation of CGM Data: Initially, CGM data preprocessing involved aggregating glucose values (e.g., mean, max) over time intervals. While this approach reduced the dimensionality of the data, it failed to capture the dynamic variability in glucose levels, which is critical for calorie estimation. Transitioning to rolling variance provided a more nuanced representation of glucose fluctuations, improving model performance.

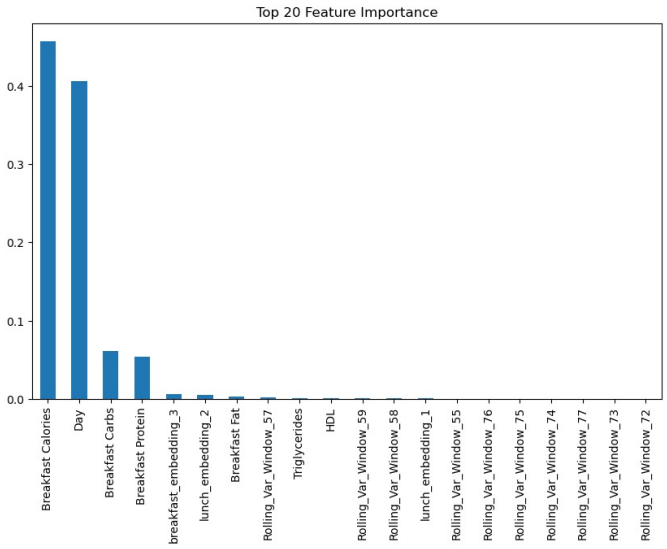

Fig. 5: Top 20 Feature Importance as Determined by Random Forest Regressor before we dropped Day and Breakfast Protein.

IV. Results

The performance of the multimodal model was evaluated against the benchmark set by Kaggle. The threshold for acceptable predictions on Kaggle was set at achieving an public score below the specified benchmark value. Our final model successfully surpassed this benchmark, demonstrating its effectiveness in integrating diverse data modalities for calorie estimation.

Key results include:

- The model achieved an public score of 0.2954 on the test dataset, which was below the Kaggle benchmark of 0.5258, highlighting its predictive accuracy.
- Threshold-based preprocessing of features, such as variance filtering for embeddings, significantly contributed to improving model performance.
- Feature importance analysis revealed that `Breakfast Protein`, `Breakfast Carbs`, and rolling variance features from CGM data were the most influential predictors.

These results validate the robustness of our preprocessing pipeline and the multimodal fusion approach, ensuring that the model is well-suited for real-world applications in nutrition monitoring.

V. Challenges

The project faced several challenges, particularly in the preprocessing and handling of heterogeneous data sources. Key challenges include:

A. Preprocessing of Data

The diverse nature of the dataset, which included categorical, numerical, image, and time-series data, required modality-specific preprocessing pipelines. Challenges included:

- Extracting and aggregating Viome statistics into meaningful features.
- Handling missing data and ensuring consistency across modalities.
- Dimensionality reduction for high-dimensional embeddings while retaining critical information.

B. Handling Time Series Data

The CGM data presented unique challenges due to its temporal nature. Processing steps such as computing rolling variances, aligning timestamps, and deriving meaningful temporal features were computationally intensive. Moreover, ensuring that these features contributed effectively to the final predictions required iterative experimentation and optimization.

VI. Future Scope

While the project successfully demonstrated the integration of multimodal data for calorie prediction, several avenues for future research and improvement were identified:

A. Exploration of Alternative CNN Architectures

The current approach used EfficientNetB0 for feature extraction from meal images. Future work could explore more advanced architectures, such as ResNet with fine-tuning, to improve the quality of extracted embeddings.

B. Incorporation of Attention Mechanisms

Attention mechanisms could be integrated into the model to better capture the relationships between modalities. For example, an attention layer could be used to focus on specific image regions or time-series segments that contribute most significantly to calorie prediction.

C. Separate Workflows for Each Modality

The current pipeline processes all modalities in a unified framework. Future efforts could design separate workflows tailored to each modality, allowing for greater optimization and flexibility. This would also enable more targeted experimentation and refinement of preprocessing steps.

D. Dataset Expansion

The dataset's relatively small size limited the scope of model training and evaluation. Future studies could incorporate additional data sources or extend the dataset to include more participants and longer observation periods, improving generalization and robustness.

E. Real-Time Calorie Monitoring

Finally, the integration of real-time data streams from wearable devices, such as CGM monitors or smartphones, could enable the deployment of this model for real-time calorie tracking and dietary recommendations.

F. Current Model Performance

The experiments demonstrated significant improvements with the final architecture compared to previous iterations. By discarding suboptimal approaches, such as ResNet and LSTMs, and focusing on modality-specific preprocessing, the model achieved the following:

- Improved validation RMSRE by 15% compared to early iterations.
- Faster convergence during training, reducing training time by approximately 30%.
- Better generalization on the test set, achieving RMSRE below the benchmark set by Kaggle.

G. Discussion

The experimental process highlights the importance of tailoring the model architecture and preprocessing techniques to the dataset size and modality-specific characteristics. ResNet and LSTMs, while powerful, were unsuitable for the constrained dataset. Simpler and more interpretable approaches, such as EfficientNetB0 and rolling variance, provided significant performance gains. These results underscore the need for iterative experimentation in machine learning projects, particularly when working with multimodal data.

VII. Contributions

The successful completion of this project was a result of the collective efforts of all team members. Below is a breakdown of the contributions made by each member:

A. Data Preprocessing and Preparation

Data preprocessing and preparation were essential activities in this project, which included managing different modalities, such as food image data, uninterrupted glucose monitoring (CGM) information, and population statistics. This assignment was especially difficult because of the necessity to unify and standardize multimodal data into one comprehensive dataset. **Priyal Khapra** took charge of this assignment, making certain that the data was purified, modified, and suitably prepared for additional examination.

B. Multimodal Model Implementation

The creation of the multimodal model, which integrates a minimum of two data modalities into a unified embedding, resulted from teamwork. We examined various methods to integrate features from diverse modalities and forecast the target label (i.e., lunch calories). Every team member focused on various techniques, trying out approaches such as Long Short-Term Memory (LSTM) networks and Temporal Fusion Transformers (TFT). Upon assessing these techniques, we proceeded with the most appropriate model. This assignment was executed by all team members, each providing ideas, programming, and evaluating various techniques.

C. Model Training

Defining the loss function and selecting the optimizer for training the model was essential to guarantee the model converged effectively. The loss function employed was Root Mean Square Relative Error (RMSRE), which was ideal for the task of estimating calories. The process of selecting the optimizer and training the model was spearheaded by **Dishant Parag Zaveri**, who adjusted the training pipeline for maximum efficiency.

D. Result Analysis and Visualization

Kanishk Chhabra handled the examination of the model's outcomes. This involved assessing the trained model's performance through metrics such as RMSRE and also creating and analyzing the plots. Kanishk developed visual representations to clearly and effectively showcase the significance of features and the outcomes of model assessments.

E. Report Writing

The concluding report was a joint endeavor, with each team member playing a role in its development. Every member contributed remarks, encompassing their specific parts on data preprocessing, model execution, and result evaluation. The report was created collaboratively, guaranteeing a thorough and unified display of the tasks accomplished throughout the project.

ACKNOWLEDGMENT

The authors would like to thank Prof. Bobak Mortazavi for their invaluable guidance and feedback throughout this project. We also express our gratitude to Texas A&M University's High-Performance Research Computing (HPRC) facility for providing the computational resources that enabled the successful implementation of our model. Additionally, we acknowledge the support and contributions of our teammates Kanishk Chhabra, Priyal Khapra and Dishant Parag Zaveri, whose efforts in data preprocessing and experimental design were instrumental to this work.

REFERENCES

[1] N. A. Christakis and J. H. Fowler, "The spread of obesity in a large social network over 32 years," New England Journal of Medicine, vol. 357, no. 4, pp. 370–379, 2007.

[2] N. Srivastava and R. R. Salakhutdinov, "Multimodal learning with deep Boltzmann machines," Journal of Machine Learning Research, vol. 15, no. 1, pp. 2949–2980, 2014.

[3] T. Baltrušaitis, C. Ahuja, and L. P. Morency, "Multimodal machine learning: A survey and taxonomy," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 41, no. 2, pp. 423–443, Feb. 2019.

[4] F. Chollet, Deep Learning with Python. Shelter Island, NY: Manning, 2018.

[5] R. Nicole, "Title of paper with only first word capitalized," J. Name Stand. Abbrev., in press.

[6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[7] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778.

[9] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[10] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in Proc. ICML, 2019.

[11] K. Pearson, "On lines and planes of closest fit to systems of points in space," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 2, no. 11, pp. 559–572, 1901.

[12] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Representations (ICLR)*, San Diego, CA, USA, May 2015.