Cardiac MRI Segmentation

Team Name: Segmentation Faults

Members: Scholtz Bálint András (A805M2), Schmieder Nándor (E9CLSH)

Introduction

This project focuses on improving the precision of semantic segmentation in cardiac MRI images by employing **ensemble** methods with multiple deep learning architectures. Utilizing the 2018 Atria Segmentation Dataset, our objective is to assess how combining various models can enhance segmentation performance compared to individual models.

Purpose

The primary aim is to investigate the effectiveness of ensemble techniques in semantic segmentation tasks. By integrating outputs from different models, we seek to achieve higher accuracy, acknowledging that this approach may require increased computational resources.

Problem

Cardiac MRI segmentation is a challenging task that requires high precision for clinical use. This project addresses the need to improve segmentation performance by constructing and analyzing **ensembles**, which can yield more reliable and accurate results than single-model solutions.

Methodology

1. Model Training: We trained several deep learning models on the 2018 Atria Segmentation Dataset, including:
   - **U-Net** [3]**:** A convolutional neural network designed for biomedical image segmentation.
   - **UNet++** [1]**:** An extension of U-Net with nested and dense skip connections to better capture multiscale features.
   - **UNETR** [2]**:** A model that incorporates transformers for 3D medical image segmentation.
   - **DynUNet** [4]: A dynamic U-Net architecture implemented within the MONAI framework.

2. Ensemble Construction: We combined the outputs of these models using a majority voting scheme to improve segmentation accuracy.
3. Performance Analysis: We evaluated the ensemble's performance, focusing on accuracy improvements and the associated computational costs.

Data Preparation

In this study, we employed a comprehensive data preprocessing strategy to prepare cardiac MRI scans and their corresponding left atrium (LA) cavity labels for model training and evaluation. The preprocessing pipeline encompassed data conversion, dataset partitioning, and organization of test data, ensuring the data's suitability for subsequent analysis.

Data Conversion

The raw MRI scans and associated LA cavity labels were converted into the Hierarchical Data Format version 5 (HDF5), resulting in the creation of the *TrainingSet.h5* file. This format facilitated efficient storage and access during the training phase. Additionally, the test data were organized into the *preprocessed_data/* directory for streamlined evaluation.

Training and Validation Split

To assess the model's generalization capability, the dataset was partitioned into training and validation subsets. Specifically, 20% of the training data was randomly selected and reserved as validation data. A fixed random seed was utilized to ensure the reproducibility of this split across different experimental runs.

Test Data Organization

The test dataset comprised 54 MRI scans, each stored as a series of .tiff files within individual folders located in the preprocessed_data/ directory. Each folder contained both the MRI slices and their corresponding LA cavity mask labels, facilitating organized access for evaluation purposes.

Summary of Preprocessing Outputs

- *TrainingSet.h5*: This file, located in the *preprocessed_data/* directory, encompassed the complete training dataset in HDF5 format, optimized for efficient training processes.
- Test Data Folders: A total of 54 directories within the *preprocessed_data/* folder, each containing .tiff files representing MRI slices and their corresponding cavity masks for the test set.

This meticulous preprocessing approach ensured that the data were appropriately formatted and partitioned, thereby enhancing the efficacy of the model training and evaluation phases.

Training Process

The training pipeline included:

1. Loss Function:

   The Dice Loss, optimized for segmentation tasks, was used to measure overlap between predictions and ground truth. A softmax activation was applied to the model outputs to calculate the loss.

2. Optimization:

   The Adam optimizer was employed, with a learning rate of 0.001. Gradients were computed and updated iteratively to minimize the loss.

3. Batch Processing:

   Training was conducted in mini-batches, with a batch size of 16.

4. Evaluation:

   Validation was performed after each training epoch to monitor the model's performance on unseen data. Metrics such as average Dice Loss were computed to identify the best-performing model.

5. Model Checkpointing:

   The best model, determined by the lowest validation loss, was saved for future use.

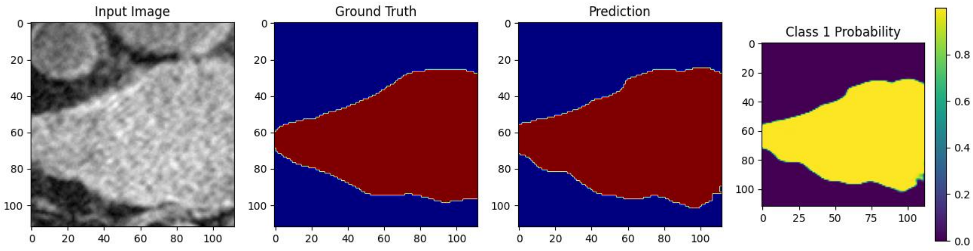


Figure2: Example of UNet++ prediction compared to ground truth

Hyper parameter Optimization and Challenges Encountered

Due to another course we were enrolled in, we were fortunate to gain access to the KIFÜ High-Performance Computing (HPC) infrastructure [5] provided by the Governmental Information Technology Development Agency. This supercomputing resource, designed for scientific and R&D tasks, enabled us to run extensive experiments efficiently and test various hyper parameter configurations with ease.

The availability of such computational power made it possible to train multiple models in parallel, significantly speeding up the exploration process. However, this advantage also introduced the potential risk of overfitting, as fine-tuning on the training data became more accessible.

To address this, we employed a rigorous trial-and-error approach, carefully monitoring validation performance to prevent over-optimization. Despite these challenges, this process ultimately resulted in the development of several robust and functional segmentation models, capable of delivering reliable results.

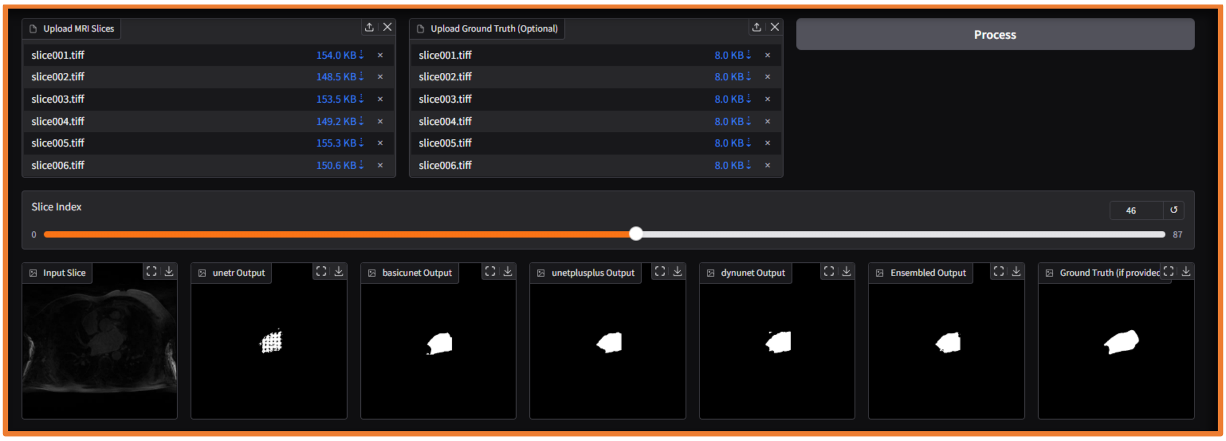
User Interface

One of the key components of this project is the interactive Gradio user interface we developed. This interface is designed to make the model outputs easily accessible and interpretable for users. As shown in the image, users can upload MRI slices and optionally provide corresponding ground truth labels.

The interface processes these inputs and displays the following outputs:

1. Input Slice: The raw MRI image uploaded by the user.
2. Model Outputs: Segmentation results generated by individual models, including UNETR, U-Net, U-Net++, and DynUNet.
3. Ensembled Output: A combined output from all models, generated using majority voting, to provide a consensus segmentation.
4. Ground Truth (if provided): For comparative evaluation, the ground truth segmentation is displayed alongside the model predictions.

A slider enables users to navigate through slices, while the interface dynamically updates all outputs for the selected slice. This design allows users to easily visualize and compare segmentation results across different models and assess their agreement with the provided ground truth. By making the outputs interactive and accessible, the interface bridges the gap between technical development and practical usability.



Upload MRI Slices

slice001.tiff 154.0 KB
slice002.tiff 148.5 KB
slice003.tiff 153.5 KB
slice004.tiff 149.2 KB
slice005.tiff 155.3 KB
slice006.tiff 150.6 KB

Upload Ground Truth (Optional)

slice001.tiff 8.0 KB
slice002.tiff 8.0 KB
slice003.tiff 8.0 KB
slice004.tiff 8.0 KB
slice005.tiff 8.0 KB
slice006.tiff 8.0 KB

Slice Index: 46

Figure 2: Gradio User Interface

Evaluation

The performance of the individual models and the ensemble was assessed using the F1 score, which measures the balance between precision and recall for segmentation tasks. The results highlight the variability in performance across different architectures, as well as the benefits and limitations of the ensemble approach.

Quantitative Results

The table below summarizes the F1 scores for each model and the ensemble:

Model	F1 Score (Mean)
U-Net	0.736
UNet++	0.599
UNETR	0.716
DynUNet	0.401
Ensemble	0.709

Insights

Among the individual models, the Basic U-Net achieved the highest F1 score of 0.736, indicating its robustness in segmenting cardiac MRI images. UNet++ closely followed with a score of 0.716, showcasing its ability to capture multiscale features effectively. However, the DynUNet and UNETR architectures underperformed, with F1 scores of 0.599 and 0.401, respectively, suggesting that their configurations may not be well-suited for this specific dataset or task.

The ensemble model, which combines predictions from all individual models using a majority voting scheme, achieved an F1 score of 0.709. While it did not surpass the performance of the Basic U-Net, the ensemble approach demonstrated a balanced performance, leveraging the complementary strengths of the individual models. However, it was unable to capitalize fully on the potential improvements that ensembling often provides, likely due to the lower performance of some constituent models.

Challenges in Evaluation

The evaluation process highlighted certain challenges, such as the variability in performance across architectures and the ensemble's sensitivity to poorly performing models. The relatively low scores of DynUNet and UNETR may have limited the ensemble's ability to outperform the best individual model. Additionally, computational costs were higher for the ensemble due to the need to aggregate predictions from multiple models.

Conclusion

This project has been an enriching experience, deepening our understanding of the complete workflow required to build machine learning solutions. From data preparation and model training to testing and evaluation, we have tackled the challenges of cardiac MRI segmentation and gained hands-on expertise in addressing them.

The development of the Gradio interface stands out as a significant achievement. By providing an accessible and interactive way to view model outputs and compare them to ground truth labels, we have ensured that our work is not only technically robust but also practically useful for researchers and clinicians alike.

We are proud of what we have accomplished through this project. It has enhanced our technical skills, reinforced our understanding of machine learning in medical imaging, and provided us with valuable insights into real-world applications. Looking forward, we are excited to build upon this foundation and explore more advanced methodologies and use cases.

Role of Language Models in This Project

Throughout the course of this project, we extensively utilized ChatGPT [6] as a valuable tool. It assisted us in crafting well-structured documentation, refining technical explanations, and streamlining our workflow. Additionally, it proved to be instrumental in debugging certain implementation challenges, offering insights and solutions that saved time and enhanced the overall quality of our work.

Bibliography

[1] Zhou, Z., Siddiquee, M. M. R., Tajbakhsh, N., & Liang, J. (2018). UNet++: A Nested U-Net Architecture for Medical Image Segmentation. arXiv preprint arXiv:1807.10165. Retrieved from https://arxiv.org/abs/1807.10165

[2] Hatamizadeh, A., Tang, Y., Nath, V., Yang, D., Myronenko, A., Landman, B., Roth, H. R., & Xu, D. (2021). UNETR: Transformers for 3D Medical Image Segmentation. arXiv preprint arXiv:2103.10504. Retrieved from https://arxiv.org/abs/2103.10504

[3] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. *arXiv preprint arXiv:1505.04597.* Retrieved from https://arxiv.org/abs/1505.04597

[4] MONAI Framework. DynUNet Implementation. Retrieved from https://docs.monai.io

[5] KIFÜ High-Performance Computing (HPC) Documentation. Retrieved from https://docs.hpc.kifu.hu

[6] ChatGPT. OpenAI. Retrieved from https://openai.com/chatgpt