WangLab Coding Test (AI hematologist)

Mohammed Baharoon

February 9, 2024

1 Methodologies

This document details the methodologies used by me to complete the coding test provided by the WangLab for a research internship at the Vector Institute. Mohammed [me :)] is very excited about this opportunity! All the code used for this work is available at: https://github.com/MohammedSB/AIHematologist.

1.1 Data Processing

In the dataset, masks are provided for the cytoplasm and nucleus of each blood cell. However, the overall goal is to segment the whole cell (cytoplasm and nucleus) and the nucleus only, not the cytoplasm alone. Because of that, I transformed the masks into whole cell and nucleus masks, by re-defining the cytoplasm mask to being cytoplasm combined with the nucleus, shaping the task as a multi-label segmentation problem. This approach has been shown to achieve better results [4], as it is more aligned with the overall task objective.

For training architectures with the nnUNet [5] framework (Sections 1.2.1 and 1.2.2), default nnUNet data augmentations were used, which include spatial transformations like rotations and scaling, and noise/visual transformations including Gaussian noise and brightness transformations. For training DINOv2 [8] (Section 1.2.3), I used random vertical and horizontal flipping and rotations.

All images were resized to 448x448. Nearest-neighbor interpolation was used for transforming masks, while bi-cubic interpolation was used for resizing blood cell images.

All input images were converted to PNG format for consistency. The images were originally in TIFF and JPG formats.

1.2 Architecture

In this section, I will discuss the two architectures used to fulfill the test's requirement. An additional bonus submission based on DINOv2 is detailed in 1.2.3.

1.2.1 Convolutional Neural Network

For the Convolutional Neural Network (CNN) [6], model I utilized the U-Net architecture [9], using the nnUNet framework. I used a five-fold cross-validation approach to determine the best configuration and used the default nnUNet hyper-parameters [5]. An ensemble of all five UNet models, each trained on one of the five training folds, was used to obtain the final predictions.

To combine the labels for cytoplasm and nucleus into a single cell label, I used the region-based segmentation provided in the nnUNet framework [5].

Additionally, I applied post-processing to the predictions, determining the best method using the nnUNet framework. The final approach applied was using the connected components algorithm on the entire cell label only.

1.2.2 Transformer-based Network

For the transformer-based network, I employed SwinUNETR [3], using the implementation in MONAI [2]. I used the nnUNet framework with SwinUNETR, using the implementation provided in [7]. Due to computational constraints, I used 100 epochs for training only and did not use five-fold cross-validation, training on the entire dataset instead. Just like in Section 1.2.1 Region-based segmentation was used.

Here, I used connected components (CC) post-processing, using the largest k, where k = 1. Before applying CC, I equalized all the labels in the predicted masks, assigning them the same value. This helped ensure that the entire cell (label 1) and nucleus (label 2) are considered as a single component. After applying CC, I reverted the targets to their original values, where a pixel would have label 2 if it had originally had label 2 before applying CC.

1.2.3 DINOv2 (Bonus)

Moreover, I utilized an additional Transformer-based network based on DINOv2 and a U-Net decoder. I used the ViT-B architecture with parameters distilled from a ViT-g pre-trained using DINOv2's self-supervised learning method on

142 million curated natural images [8]. I used a U-Net decoder made up of four blocks, where each block consists of one convolutional layer along with ReLU activation function and batch normalization, as is done in [1]. Skip connections were obtained from the block numbers 2, 4, 7, 9, 12 of the transformer model and concatenated to the features at each U-Net layer, as is done in the classical U-Net architecture.

I used a learning rate of 1-e2 for fine-tuning the newly initialized decoder, and a learning rate of 1e-5 for the pre-trained weights in the DINOv2 encoder. The learning rate for the encoder was determined by scanning the values {5e-4, 1e-4, 5e-5, 1e-5, 5e-6, 1e-6, 5e-7, 1e-7}, each for 100 validation epochs. I trained the model for 1000 epochs.

Both the dice and the combined dice and cross-entropy losses were evaluated. Qualitatively, the dice loss resulted in more stable training, so I used it for the final training.

The same post-processing as described in Section 1.2.2 was applied.

References

[1] Mohammed Baharoon, Waseem Qureshi, Jiahong Ouyang, Yanwu Xu, Abdulrhman Aljouie, and Wei Peng. Towards general purpose vision foundation models for medical image analysis: An experimental study of dinov2 on radiology benchmarks, 2023.

[2] M. Jorge Cardoso, Wenqi Li, Richard Brown, Nic Ma, Eric Kerfoot, Yiheng Wang, Benjamin Murrey, Andriy Myronenko, Can Zhao, Dong Yang, Vishwesh Nath, Yufan He, Ziyue Xu, Ali Hatamizadeh, Andriy Myronenko, Wentao Zhu, Yun Liu, Mingxin Zheng, Yucheng Tang, Isaac Yang, Michael Zephyr, Behrooz Hashemian, Sachidanand Alle, Mohammad Zalbagi Darestani, Charlie Budd, Marc Modat, Tom Vercauteren, Guotai Wang, Yiwen Li, Yipeng Hu, Yunguan Fu, Benjamin Gorman, Hans Johnson, Brad Genereaux, Barbaros S. Erdal, Vikash Gupta, Andres Diaz-Pinto, Andre Dourson, Lena Maier-Hein, Paul F. Jaeger, Michael Baumgartner, Jayashree Kalpathy-Cramer, Mona Flores, Justin Kirby, Lee A. D. Cooper, Holger R. Roth, Daguang Xu, David Bericat, Ralf Floca, S. Kevin Zhou, Haris Shuaib, Keyvan Farahani, Klaus H. Maier-Hein, Stephen Aylward, Prerna Dogra, Sebastien Ourselin, and Andrew Feng. Monai: An open-source framework for deep learning in healthcare, 2022.

[3] Ali Hatamizadeh, Vishwesh Nath, Yucheng Tang, Dong Yang, Holger Roth, and Daguang Xu. Swin unetr: Swin transformers for semantic segmentation of brain tumors in mri images, 2022.

[4] Fabian Isensee, Philipp Kickingereder, Wolfgang Wick, Martin Bendszus, and Klaus H. Maier-Hein. No new-net, 2019.

[5] Fabian Isensee, Jens Petersen, Andre Klein, David Zimmerer, Paul F. Jaeger, Simon Kohl, Jakob Wasserthal, Gregor Koehler, Tobias Norajitra, Sebastian Wirkert, and Klaus H. Maier-Hein. nnu-net: Self-adapting framework for u-net-based medical image segmentation, 2018.

[6] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

[7] Jun Ma, Feifei Li, and Bo Wang. U-mamba: Enhancing long-range dependency for biomedical image segmentation, 2024.

[8] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2024.

[9] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.