Brain MRI Image Segmentation for Anomaly Detection Using U-Net

Nicolas Abule
Department of Computer Science
Toronto Metropolitan University
Toronto, Canada
nabule@torontomu.ca

Azan Haider
Department of Computer Science
Toronto Metropolitan University
Toronto, Canada
azan.haider@torontomu.ca

Jason Nguyen
Department of Computer Science
Toronto Metropolitan University
Toronto, Canada
j19nguyen@torontomu.ca

Jordan Nguyen
Department of Computer Science
Toronto Metropolitan University
Toronto, Canada
jordan1.nguyen@torontomu.ca

Enes Polat
Department of Computer Science
Toronto Metropolitan University
Toronto, Canada
epolat@torontomu.ca

Abstract—Brain segmentation and anomaly detection are important tasks in medical imaging, allowing for more accurate diagnosis and treatment planning for conditions affecting the brain. This paper investigates the application of a U-Net based deep learning architecture to automate this process. While U-Nets are known for their ability to extract fine and global contextual features, the results of this study show significant room for improvement. The model demonstrated middling performance in segmenting brain regions and detecting anomalies, which shows the challenges posed by the variability in imaging data. This project shows the need for future research to explore more advanced solutions and better training. Ultimately, this project lays the foundation for using U-Net architecture, which can be expanded upon. Our code is available at https://github.com/JasonVTNguyen/CPS843_BrainSegmentation.

I. INTRODUCTION

The accurate segmentation of brain images and identification of anomalies such as tumors, lesions, or other features play a critical role in the fields of diagnostics, psychology, and neuroscience. Advances in the techniques used in neuroimaging, such as Magnetic Resonance Imaging (MRI) and Computed Tomography (CT), have allowed for the collection of high resolution datasets. However, manual segmentation of these images is a time-consuming, labour-intensive, and subjective task which is prone to inter-observer and intra-observer variability. Given these challenges, the need for automated methods that are both efficient and reliable has become all the more important.

Brain segmentation involves partitioning an image into different meaningful regions, such as grey matter, white matter, and cerbrospinal fluid (CRF) [1]. This step is important for many tasks that follow, such as surgical planning and disease progression monitoring. Further, the ability to detect anomalies in brain images is crucial for early diagnosis and more personalized treatment planning for certain conditions.

Recent advances in deep learning have transformed the field of medical image analysis, providing state-of-the-art performance in segmentation, classification, detection, and image registration [2]. Convolutional Neural Networks (CNNs) have shown great promise in extracting complex features from these brain images. These models, alongside good preprocessing and postprocessing, show a way for achieving high accuracy as well as clinical usage.

Our contribution in this project is the enhanced customizability in the training process allowing for specific seeds and batch sizes to be used. This allows for reproducibility in the results achieved while allowing for controlled experimentation by choosing different seeds and different batch sizes. This allows for users to optimize their training processes based on their computational resources and or on their dataset.

II. BACKGROUND

U-Net - The U-Net is a deep learning model designed for image segmentation tasks and has become a cornerstone in medical image analysis due to its ability to achieve precise pixel-level segmentation. First introduced in [3], the U-Net architecture is characterized by its contracting-expanding structure. The contracting path follows the typical architecture of a convolutional network. This consists of repeatedly doing convolutions, then doing a rectified linear unit (ReLU) as well as a max pooling operation for downsampling, with each downsampling doubling the number of features. The expanding path does the opposite by upsampling the feature map, then performing a convolution which halves the amount of features, then concatenates it with the cropped feature map. An important feature of U-Net is the inclusion of skip connections, which allow for directly linking the corresponding layers of the different paths. These connections enable the model to retain and reuse spatial information that would have been lost during the process of downsampling.

U-Net's ability to capture global context as well as complex details make it particularly effective for brain segmentation, where the accurate seperation of strucutres of grey matter,

white matter and CSF is essential. This paper uses a U-Net based model to segment the brain's structures to find anomalies.

III. METHODOLOGY

The implementation of this project uses many different libraries and tools to allow the training and evaluation of our U-Net model. The center of this training is TensorFlow which allows for the building and training of the neural network. The U-Net architecture is being used from the library unet specifically build_unet. The Adam optimizer is also used as it offers model optimization and efficient gradient-based parameter updates to ensure stable convergence during training.

The dataset that we used contains both MRI images as well as masks that correspond to the anomaly location for any given image which allows us to compare our model's output to the given mask, this dataset can be found at [4]. The images from the dataset we utilized is sourced from [5].

In order to generate a model for segmentation, U-Net was used to build a neural network that would take and compare the brain scan images and the masks from the dataset. U-Net architecture is designed to provide better accurate segmentations with less training images. The architecture works similar to a traditional neural network using consecutive layers, but upscales the images in between layers assembling a precise output. This U-Net model is stored in a python file 'unet.py' and runs via an imported build_unet function. For better control over the program, and to avoid having to train the model after every instance of running the program, training and testing are separated into two different python files: train.py and test.py.

Train.py is the program that trains the neural network for prediction. First, the dataset is split into three sets: a training set, a test set, and a validation set. The X, or the input are the brain scans, and the y, the target class or output, are the masks that specifically indicate the location of the tumor in the input image. The training set is compiled on the U-Net model developed from unet.py, using three user-defined variables required at the beginning of running the python program: seed, number of epochs, and number of batches. The seed defines a value to initialize a random number generator, which is important when randomly splitting the dataset into the previously mentioned sets. Number of epochs define the amount of epochs the training lasts for, that being the more epochs will make a more accurate model, however it takes more time to train the program. Number of batches defines the amount of batches the training will process at a time, higher numbers will process quicker at the cost of demanding more resources from the machine. Once the model has been trained, the user can run the testing program test.py. This also generates a file called memory.txt which will be required for test.py.

Test.py provides important information and outputs the resultant images from the prediction. It reads memory.txt, which contains the seed that is important for parity between the training and testing program. The number of epochs, purely for visual representation of the comprehensiveness of the model.
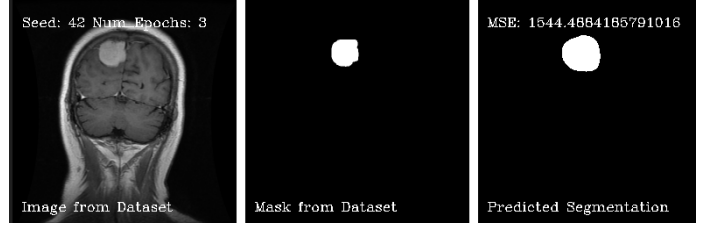

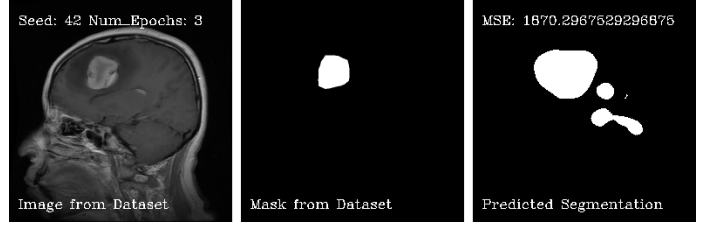Fig. 1. Much larger area segmented for the anomaly shown

Fig. 2. Segmenting areas that are not a part of the anomaly

Running test.py will load the model and dataset and will start to consecutively predict a segmentation based on the inputted image. This will output a prediction and stored in variable y_pred. For easy accessibility, the result output will be a spread of three images: the original brain scan image, the mask derived from the dataset, and the predicted segmentation. Information concerning accuracy will be output as an average in the console, and individually in a csv file.

IV. RESULTS

Looking at the segments that the model creates often does cover the mask shown in the dataset, but often creates a much larger area then required (as seen in figure 1) and often covers areas which do not have any anomalies (as seen in figure 2).

Also, the performance of our project was evaluated using four metrics: F1 Score, Jaccard Score, precision, and recall. The results of the project are as follows:

A. F1 Score

The model achieved an average F1 score of 0.28379. The highest F1 score achieved is 0.93689, with the lowest score achieved being 0.0. This in absolute terms is a poor score, it shows that while the project will sometimes be right it will often also flag false positives as well as false negatives. There is much improvement needed in terms of decreasing both of those values.

B. Jaccard Score

The model achieved an average Jaccard score of 0.20459. The highest Jaccard score achieved is 0.88127, with the lowest score achieved being 0.0. This suggests that there is some overlap in our predicted mask as compared to the mask given by the dataset, but this overlap is small, showing that the model's segmentation accuracy is low.

C. Precision

The model achieved an average precision of 0.27867. The highest precision achieved is 1.0, with the lowest score achieved being 0.0. This shows that while there are many true positives that are detected properly, the model still creates a large number of false positives.

D. Recall

The model achieved an average recall of 0.44643. The highest recall achieved is 1.0, with the lowest score achieved being 0.0. This shows that the identification of false negatives is far lower than the rate of which the model finds false positives. But this score is still not high meaning the amount of false negatives is still relatively high.

V. DISCUSSION

The results that we got in this project show the potential of the U-Net approach that we used for brain segmentation and anomaly detection. The performance metrics we use, while not all encompassing, do show the effectiveness of our model which is generally poor. The low F1 score and Jaccard score suggest that the model cannot properly get a good precision and recall while also having a good overlap with the ground truth masks provided by the dataset. This means that the model is both oversegmenting and undersegmenting depending on what image it is evaluating. Improving the alightment between the predicted and ground truth masks is an area for improvement.

This model did suffer from a lack of training as it was trained for around 3 hours, reaching only its third epoch. Such limited training time likely contributed to the low performance metrics observed, as the model may not have fully learned the underlying patterns and features necessary for accurate segmentation. Deep learning models, particularly U-Net architectures, often require extensive training over many epochs to optimize their parameters.

The dataset used for training, while adequate for preliminary testing, may not provide sufficient variety or balance in the types of anomalies it represents. This limitation can result in an overfit model that performs well on the training data, but struggles to generalize to new, unseen images. Another key limitation is the absence of post-processing techniques that could refine the segmentation masks. Implementing such techniques could help correct oversegmentation and undersegmentation issues, leading to more accurate predictions.

In future projects, several improvements could be made to improve the performance of the model. First, increasing the length of training and altering batch sizes could help with getting more accurate results as well as possibly using less time for said results. Second, looking at more types of architectures to combine with this one could also inprove segmentation accuracy. Thirdly, adding post processing could improve the segmentation results which could lead to less oversegmentation and or undersegmentation. Lastly, utilizing a larger dataset and/or a more diverse dataset could also improve the results that we have achieved.

VI. CONCLUSION

This paper explored the usage of U-Net based architecture for brain segmentation and anomaly detection. While the results obtained from our project show that there is much improvement to be made, the study shows that there is potential for deep learning models in addressing the tasks in medical imaging. The challenges faced highlight the need for more techniques to improve the segmentation accuracy and optimization in training processes.

VII. FUTURE WORK

While this project demonstrates the effectiveness of U-Net-based models for brain segmentation and anomaly detection, there remain several opportunities for future research. One of these opportunities is that of using more advanced architecture, such as transformers and hybrid models which could combine convolutional and attention mechanisms. Another improvement could come from using multi-model data, such as combining MRI and CT scans, which could improve the robustness of the segmentation and allow for more fine details to be detected where they might not have been given a singular data type.

Multi-task learning is also another avenue for improving the effectiveness by broadening the scope of what these models are able to do. Through designing models that don't just detect, but also classify anomalies or predict disease progression, much more will be able to be done with the same data. Furthermore, a shared training process can make the model more generalizable as it makes the model extract features that would be applicable to multiple tasks and not just the one.

As machine learning models are being increasingly used in clinical practice, it is important to address the bias that these models may show. Models that are trained on non-representative datasets risk introducing variability in performance across different demographic groups, which could potentially lead to inequitable healthcare outcomes. For example, a lack of women in the dataset for MRIs may lead to a lower accuracy score for conditions that would be more prevalent for women. Future work in this area should be in crafting datasets that are diverse and representative of the population to allow for equitable performance.

VIII. CONTRIBUTIONS

Contributions by group members are as follows: The code was written by Jason Nguyen. The report was primarily written and researched by Nicolas Abule. The methodology being primarily written by Jason Nguyen, with contributions by Nicolas Abule. The report was proofread and edited by Jordan Nguyen. The installation guide and video demo were made by Jordan Nguyen. Other miscellaneous tasks were completed by Nicolas Abule and Jordan Nguyen.

REFERENCES

[1] Wang, Y., Wang, Y., Zhang, Z., Xiong, Y., Zhang, Q., Yuan, C., & Guo, H. (2018). Segmentation of gray matter, white matter, and CSF with fluid and white matter suppression using MP2RAGE. Journal of magnetic resonance imaging : JMRI, 48(6), 1540–1550. https://doi.org/10.1002/jmri.26014

[2] Chen, X., Wang, X., Zhang, K., Fung, K. M., Thai, T. C., Moore, K., ... & Qiu, Y. (2022). Recent advances and clinical applications of deep learning in medical image analysis. Medical image analysis, 79, 102444.

[3] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18 (pp. 234-241). Springer International Publishing.

[4] Tomar, N. (2022). Brain Tumor Segmentation. https://www.kaggle.com/datasets/nikhilroxtomar/brain-tumor-segmentation

[5] Cheng, J. (2017). brain tumor dataset (Version 5). figshare. https://doi.org/10.6084/m9.figshare.1512427.v5