Rapport de Stage

Préparé par : Mohamed naouar ING 1 info génie logiciel

Introduction

Le stage effectué au sein du laboratoire de médecine imagerie médicale a été une expérience passionnante et enrichissante dans le domaine de la médecine et de l'informatique. Au cours de ce stage, j'ai eu l'opportunité de me plonger dans le monde fascinant du traitement d'images médicales, plus précisément dans le domaine de l'imagerie cérébrale. Mon stage a été centré sur le développement d'architectures de réseaux de neurones convolutifs (CNN) pour la segmentation d'images cérébrales, ainsi que sur la création d'une interface graphique pour intégrer ces modèles dans l'application du laboratoire.

Objectif du Stage

L'objectif fondamental de mon travail était de parvenir à une segmentation précise des images cérébrales, en identifiant les différentes structures et régions d'intérêt dans le cerveau. Pour ce faire, j'ai utilisé l'architecture U-Net, une approche de pointe dans le domaine de la segmentation sémantique. Le modèle U-Net est reconnu pour sa capacité à capturer les caractéristiques spatiales et contextuelles importantes des images, ce qui en fait un choix idéal pour les tâches de segmentation d'images médicales.

Préparation des Données

Le processus de développement du modèle U-Net a impliqué plusieurs étapes clés. Tout d'abord, j'ai préparé les données d'entraînement en utilisant des bibliothèques telles que PIL, OpenCV et NumPy pour le chargement, le redimensionnement et la normalisation des images. J'ai également préparé les masques correspondants pour l'apprentissage supervisé du modèle.
Pendant le stage, une étape cruciale a été la création de la vérité terrain pour les images de cerveau. La vérité terrain (ground truth) fait référence aux annotations manuelles ou aux masques binaires précisément définis pour chaque image cérébrale, identifiant ainsi les différentes structures d'intérêt telles que les lobes frontaux, pariétaux, temporaux et occipitaux.
Pour réaliser cette phase de vérité terrain, j'ai utilisé un logiciel nommé Amira, qui est largement utilisé dans le domaine de la recherche en imagerie médicale. À l'aide d'Amira, j'ai pu parcourir chaque image cérébrale et effectuer des annotations manuelles pour identifier et segmenter les régions cérébrales spécifiques.
Les résultats de cette phase de vérité terrain ont été essentiels pour l'entraînement et l'évaluation du modèle U-Net de segmentation d'images cérébrales. En fournissant des masques binaires précis pour chaque image, la vérité terrain a permis de créer un ensemble de données d'entraînement supervisé pour guider le modèle lors de son apprentissage. Ces masques ont joué un rôle déterminant dans la qualité du modèle en fournissant des informations précieuses pour l'apprentissage des caractéristiques importantes des différentes régions du cerveau.

Implémentation du Modèle U-Net

Ensuite, j'ai mis en œuvre l'architecture U-Net à l'aide de Keras et TensorFlow, en utilisant une fonction de coût basée sur le coefficient Dice pour entraîner le modèle. Le coefficient Dice est une métrique couramment utilisée pour évaluer la qualité des segmentations binaires, en mesurant la similitude entre les régions prédites et les régions réelles dans les images.

Entraînement et Évaluation du Modèle

Une fois que le modèle a été entraîné avec succès sur un ensemble de données d'images cérébrale j'ai évalué ses performances en utilisant l'indice IoU (Intersection over Union) pour mesurer l'adéquation des prédictions par rapport aux masques réels. J'ai également réalisé une inspection visuelle en comparant les images d'entrée, les masques réels et les prédictions.

Voici le résumé de l'architecture du modèle U-Net avec les dimensions de chaque couche :

Le modèle U-Net est composé de plusieurs couches de convolutions, de max pooling et de transposition qui permettent d'extraire des caractéristiques à différentes échelles. Voici les dimensions de chaque couche du modèle :

1. **Couche d'entrée (Input Layer) : (None, 320, 320, 1)**
2. **Conv2D_1 : (None, 320, 320, 32)**
3. **Conv2D_2 : (None, 320, 320, 32)**
4. **BatchNormalization_1 : (None, 320, 320, 32)**
5. **MaxPooling2D_1 : (None, 160, 160, 32)**
6. **Dropout_1 : (None, 160, 160, 32)**
7. **Conv2D_3 : (None, 160, 160, 64)**
8. **Conv2D_4 : (None, 160, 160, 64)**
9. **BatchNormalization_2 : (None, 160, 160, 64)**
10. **MaxPooling2D_2 : (None, 80, 80, 64)**
11. **Dropout_2 : (None, 80, 80, 64)**
12. **Conv2D_5 : (None, 80, 80, 128)**
13. **Conv2D_6 : (None, 80, 80, 128)**
14. **BatchNormalization_3 : (None, 80, 80, 128)**
15. **MaxPooling2D_3 : (None, 40, 40, 128)**
16. **Dropout_3 : (None, 40, 40, 128)**
17. **Conv2D_7 : (None, 40, 40, 256)**
18. **Conv2D_8 : (None, 40, 40, 256)**
19. **BatchNormalization_4 : (None, 40, 40, 256)**
20. **MaxPooling2D_4 : (None, 20, 20, 256)**
21. **Dropout_4 : (None, 20, 20, 256)**
22. **Conv2D_9 : (None, 20, 20, 512)**
23. **Conv2D_10 : (None, 20, 20, 512)**
24. **BatchNormalization_5 : (None, 20, 20, 512)**
25. **MaxPooling2D_5 : (None, 10, 10, 512)**
26. **Dropout_5 : (None, 10, 10, 512)**
27. **Conv2D_11 : (None, 10, 10, 1024)**
28. **Conv2D_12 : (None, 10, 10, 1024)**
29. **Conv2DTranspose_1 : (None, 20, 20, 512)**
30. **Concatenate_1 : (None, 20, 20, 1024)**
31. **BatchNormalization_6 : (None, 20, 20, 512)**
32. **Conv2D_13 : (None, 20, 20, 512)**
33. **Conv2D_14 : (None, 20, 20, 512)**
34. **BatchNormalization_7 : (None, 20, 20, 512)**
35. **Conv2DTranspose_2 : (None, 40, 40, 256)**
36. **Concatenate_2 : (None, 40, 40, 512)**
37. **BatchNormalization_8 : (None, 40, 40, 256)**
38. **Conv2D_15 : (None, 40, 40, 256)**
39. **Conv2D_16 : (None, 40, 40, 256)**
40. **BatchNormalization_9 : (None, 40, 40, 256)**
41. **Conv2DTranspose_3 : (None, 80, 80, 128)**
42. **Concatenate_3 : (None, 80, 80, 256)**
43. **BatchNormalization_10 : (None, 80, 80, 128)**
44. **Conv2D_17 : (None, 80, 80, 128)**
45. **Conv2D_18 : (None, 80, 80, 128)**
46. **BatchNormalization_11 : (None, 80, 80, 128)**
47. **Conv2DTranspose_4 : (None, 160, 160, 64)**
48. **Concatenate_4 : (None, 160, 160, 128)**
49. **BatchNormalization_12 : (None, 160, 160, 64)**
50. **Conv2D_19 : (None, 160, 160, 64)**
51. **Conv2D_20 : (None, 160, 160, 64)**
52. **BatchNormalization_13 : (None, 160, 160, 64)**
53. **Conv2DTranspose_5 : (None, 320, 320, 32)**

1. Concatenate_5 : (None, 320, 320, 64)
2. BatchNormalization_14 : (None, 320, 320, 32)
3. Conv2D_21 : (None, 320, 320, 32)
4. Conv2D_22 : (None, 320, 320, 32)
5. Conv2D_23 : (None, 320, 320, 1)

Total params: 7,765,281
Trainable params: 7,762,401
Non-trainable params: 2,880
Le modèle U-Net comprend un total d'environ 7,7 millions de paramètres, dont la majorité provient des poids de convolution pour apprendre les caractéristiques pertinentes. Chaque couche a une taille spécifique en fonction du nombre de canaux, de la hauteur et de la largeur de l'image. Le modèle a été entraîné avec succès sur les données d'entraînement et évalué sur un ensemble de test, obtenant un indice de similarité Jaccard (IoU) d'environ 0,78, ce qui indique de bonnes performances de segmentation pour les structures cérébrales.

1. Couche Convolution 1 (conv2d_38) :
   - Type de couche : Conv2D
   - Sortie (Output Shape) : (None, 320, 320, 32)
   - Paramètres (Param #) : 320
   - Description : Cette couche utilise un filtre de convolution de taille 3x3 pour extraire 32 caractéristiques différentes de l'image d'entrée. La sortie est une image avec 32 canaux, chaque canal représentant une caractéristique particulière de l'image.
2. Couche Max Pooling 1 (max_pooling2d_8) :
   - Type de couche : MaxPooling2D
   - Sortie (Output Shape) : (None, 160, 160, 32)
   - Description : Cette couche effectue une opération de max pooling avec un filtre de taille 2x2 sur l'image d'entrée. Elle réduit la taille de l'image en sous-échantillonnant les pixels avec la valeur maximale dans chaque région de 2x2. Cela permet de capturer les caractéristiques les plus importantes tout en réduisant la dimension de l'image.
3. Couche Convolution Transposée 1 (conv2d_transpose_8) :
   - Type de couche : Conv2DTranspose
   - Sortie (Output Shape) : (None, 40, 40, 256)
   - Paramètres (Param #) : 524544
   - Description : Cette couche effectue une opération de transposition en utilisant un filtre de taille 2x2. Elle augmente la taille de l'image en effectuant une interpolation et en ajoutant des valeurs de pixels intermédiaires. Cette opération est utilisée pour revenir à une résolution plus élevée tout en préservant les caractéristiques apprises à partir des couches de convolution précédentes.

   - Dans le contexte de la segmentation d'images cérébrales, un IoU de 0.7839 signifie que, en moyenne, environ 78,39 % des pixels correctement prédits par le modèle se trouvent dans la zone d'intérêt réelle du masque de segmentation. Cela indique que le modèle a réussi à capturer et à aligner efficacement les structures cérébrales d'intérêt, telles que les lobes frontaux, pariétaux, temporaux et occipitaux.
   - Un IoU élevé indique une bonne correspondance entre les prédictions du modèle et les masques réels de segmentation, ce qui suggère que le modèle est capable de bien détecter et segmenter les structures cérébrales d'intérêt dans les images. Cependant, il convient de noter que l'IoU seul ne fournit pas une évaluation complète de la performance du modèle. D'autres métriques telles que la précision, le rappel et la courbe ROC peuvent également être considérées pour une évaluation plus complète.
-

Conclusion et Perspectives d'Avenir

Ce stage a été une occasion inestimable de combiner mes compétences en informatique et ma passion pour la médecine, tout en contribuant au développement d'outils avancés pour l'analyse d'images médicales. J'espère que ce rapport sera un témoignage fidèle de mes efforts et de mes accomplissements au cours de cette expérience enrichissante. Pour l'avenir, des améliorations pourraient être apportées au modèle en optimisant les hyperparamètres et en explorant d'autres types d'images médicales. Ce projet ouvre également la voie à de nouvelles applications dans le domaine médical, contribuant ainsi à l'amélioration des diagnostics et des traitements pour les patients.