Improving brain tumor segmentation using synthetic images from diffusion models

The BraTS 2021 dataset contains MRI volumes and tumor annotations from 1251 tumor patients, see an example below. The image patches show from left to right: the whole tumor (yellow) visible in T2-FLAIR (Fig.A), the tumor core (red) visible in T2 (Fig.B), the enhancing tumor structures (light blue) visible in T1Gd, surrounding the cystic/necrotic components of the core (green) (Fig. C). The segmentations are combined to generate the final labels of the tumor sub-regions (Fig.D): edema (yellow), non-enhancing solid core (red), necrotic/cystic core (green), enhancing core (blue).

En bild som visar foto, olika, visa, tittar

Automatiskt genererad beskrivning

Using deep learning, it is possible to train a segmentation network to perform segmentation of the different parts of the tumor, but deep learning requires large datasets. Diffusion models can be trained to synthesize realistic images (see for example thispersondoesnotexist.com, which uses a GAN). In this master thesis, the main goal is to investigate if synthetic brain images, and tumor annotations, can improve training of segmentation networks. Existing Python / Tensorflow code for diffusion models will be used for synthesizing multi-channel images that contain 4 MR images as well as tumor annotations. A computer with a 6 core CPU, 64 GB RAM and 2 x Nvidia RTX 2080 Ti graphics cards is available for the project.

The following questions are of interest

Does a segmentation network trained on synthetic images perform well on real images?

How do diffusion models compare to GANs when generating realistic images?

Does a segmentation network trained on real + synthetic images perform better compared to a segmentation network only trained on real images?

How sensitive is the generation of synthetic MR images + annotations to the hyperparameters of the diffusion model? Do some settings result in synthetic images that are better for training the segmentation network?

Requirements: Python programming, Deep learning.

Contact person: [anders.eklund@liu.se](mailto:anders.eklund@liu.se)