Multimodal Brain Tumor Lesion Segmentation using Limited Labeled Images

Dhruv Sharma, Zahil Shanis, Chandan K. Reddy, Samuel Gerber, and **Andinet Enquobahrie**

VIRGINIA TECH™

Kitware

Kitware

Collaborative software R&D

- Technical computing
- Algorithms & applications
- Software process & infrastructure
- Support & training
- Open source leadership

Supporting all sectors

- Industry, government & academia

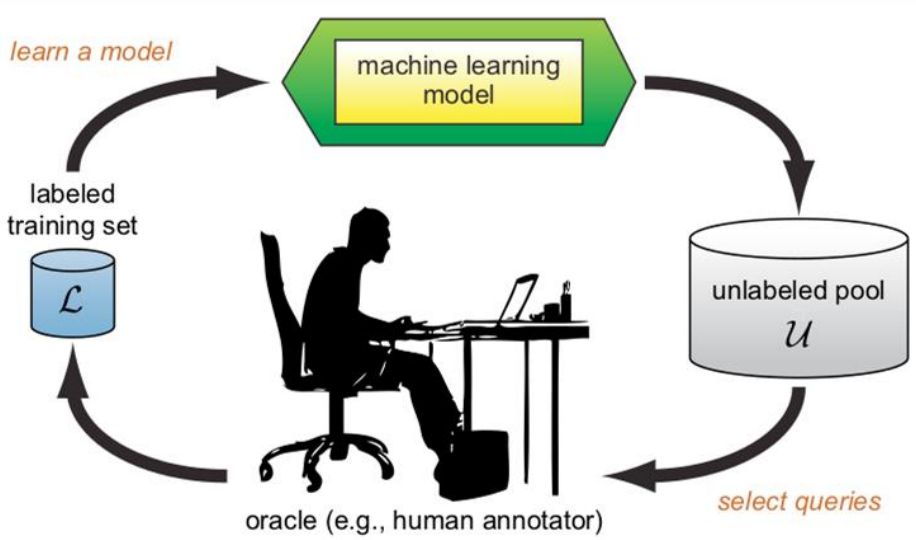
Challenges

Large amount of labeled data required to train deep learning models

Limited Medical experts to label the data

Kitware

Active Learning

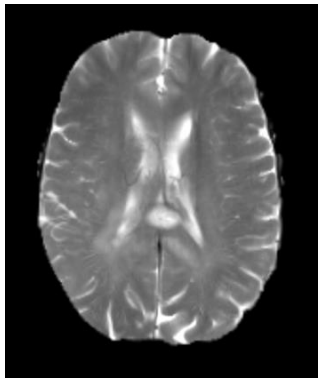


- No need to get the entire dataset labeled from expert
- Get only the most ***informative*** data points labeled and train on them
- *Informative points* - points from the unlabeled pool which impart the highest learning to a model

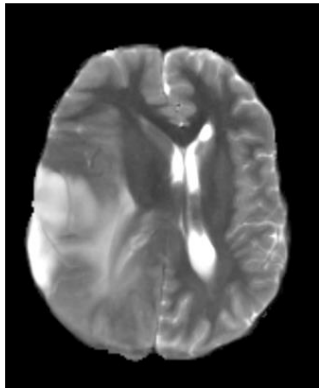
Kitware

Brain Tumor - Clinical Motivation

- Gliomas are the most common type of brain tumors which emerge from glial cells
- The gliomas can be of two types based on their severity - Low Grade Gliomas (LGG) and High Grade Gliomas (HGG)
- It becomes highly necessary to detect the Gliomas in the early stages itself
- Thus, having softwares which can segment out the tumor lesions with great accuracy are of great importance



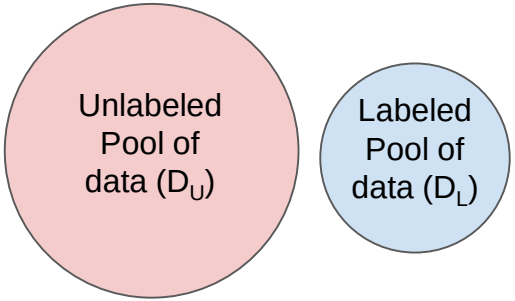
T2 scan of HGG



T2 scan of LGG

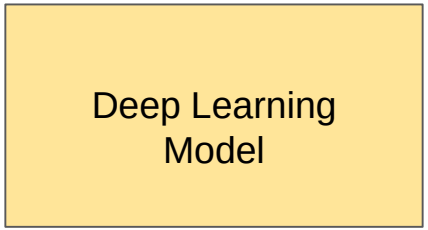
Kitware

Major Components



Pools of data - labeled and unlabeled

Data - 2018 BraTS



For the task of lesion segmentation

Model - UNet

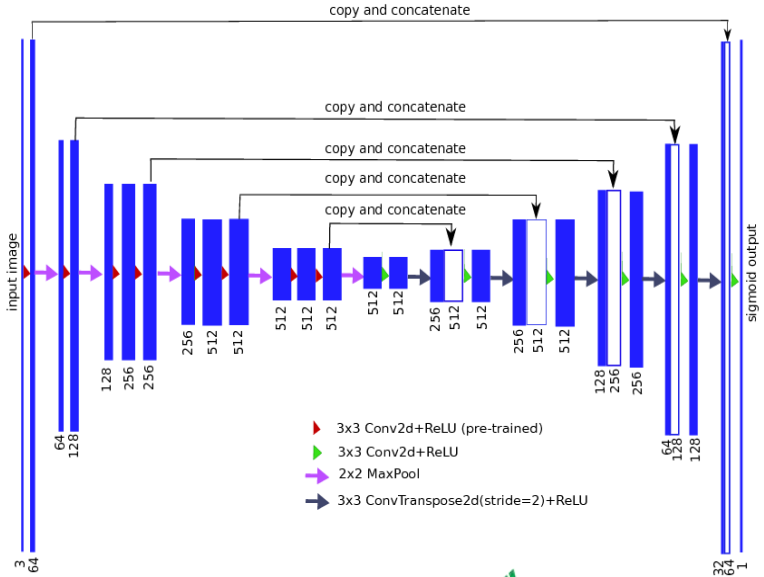
To select the most informative points

Query Strategy - Coreset Based Ranked Batch Mode Sampling

Kitware

Model Architecture

- U-Net - Classic encoder-decoder model
- encoder captures the spatial information into a reduced form using the CNN
- Decoder captures the semantic information by combining the encoder feature maps with the decoder using skip connections
- Dice loss function was used. Aims at maximizing the dice coefficient metric, thus performing better for data with class imbalance

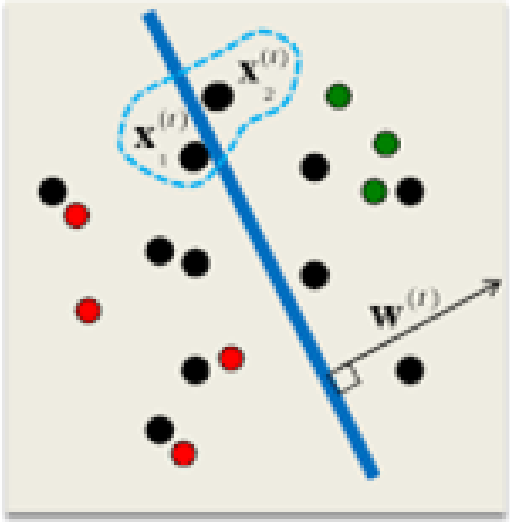


Kitware

Query Strategies

Uncertainty Sampling

- Learner queries the instances from the unlabeled pool about which it is least certain
- Certainty??? (for classification)
  - Least confident:
    - Picks the points which has the least confidence scores
    - Eg : a = [0.9, 0.1], b = [0.6, 0.4] => picks b
- Issues:
  - Does not captures the *representativeness* factor
  - Generally used with traditional ML models, and not with deep learning models

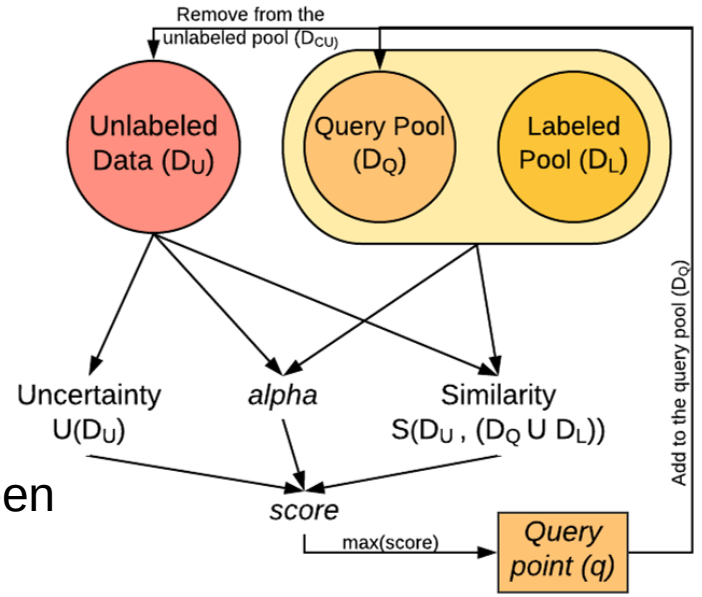


Kitware

Query Strategies

Ranked Batch-Mode Sampling

- Uncertainty:
  - Uncertainty Sampling
- Representativeness:
  - Intra-diversity
  - Inter-diversity
- Alpha maintains a balance in between the two scores
- Iterative builds the query pool



$$alpha = \frac{|D_U|}{|D_U| + |D_L U D_Q|}$$

$$score = alpha * (1 - s(D_U, D_Q U D_L) + (1 - alpha) * U(D_U)$$

Query Strategies

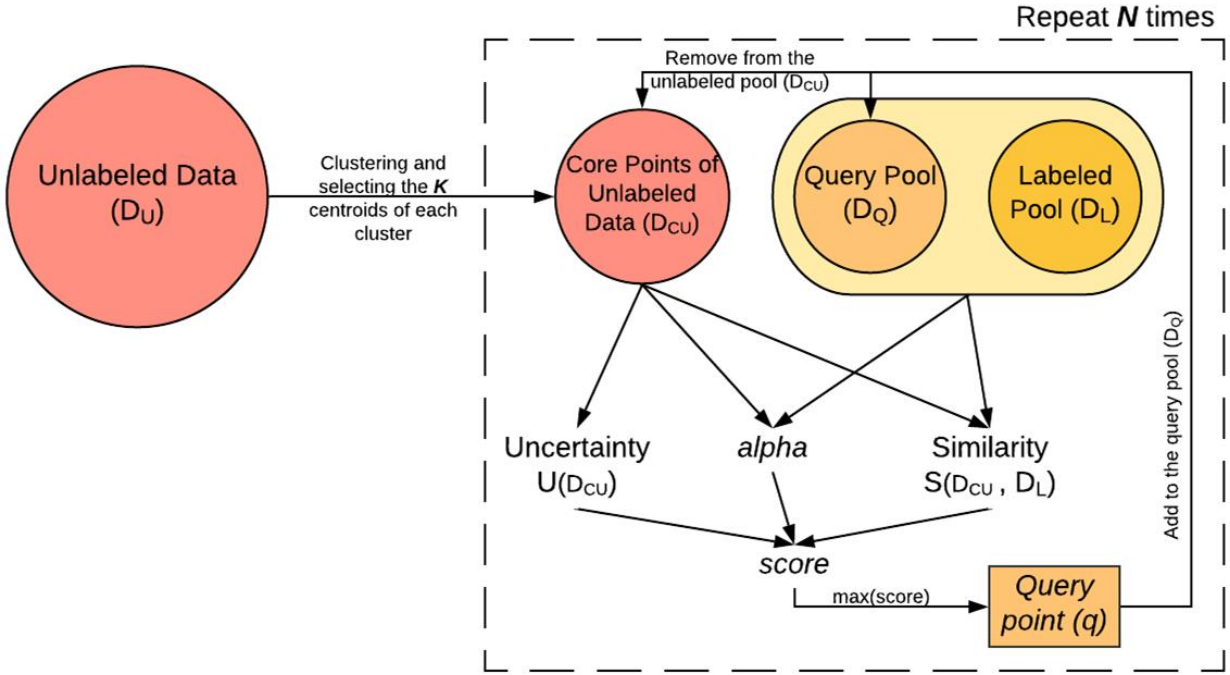
Issues with Ranked Batch Mode Sampling:

- Too much computational time and space if the dataset is too large
- Thus, increases the query time tremendously

Coreset based Ranked Batch-Mode Sampling

- Uncertainty:
  - Uncertainty Sampling
- Representativeness:
  - Intra-diversity: K-Means Clustering
  - Inter-diversity b/w reduced and labeled pool
- Alpha maintains a balance in between the two scores
- Iteratively builds the query pool

Kitware

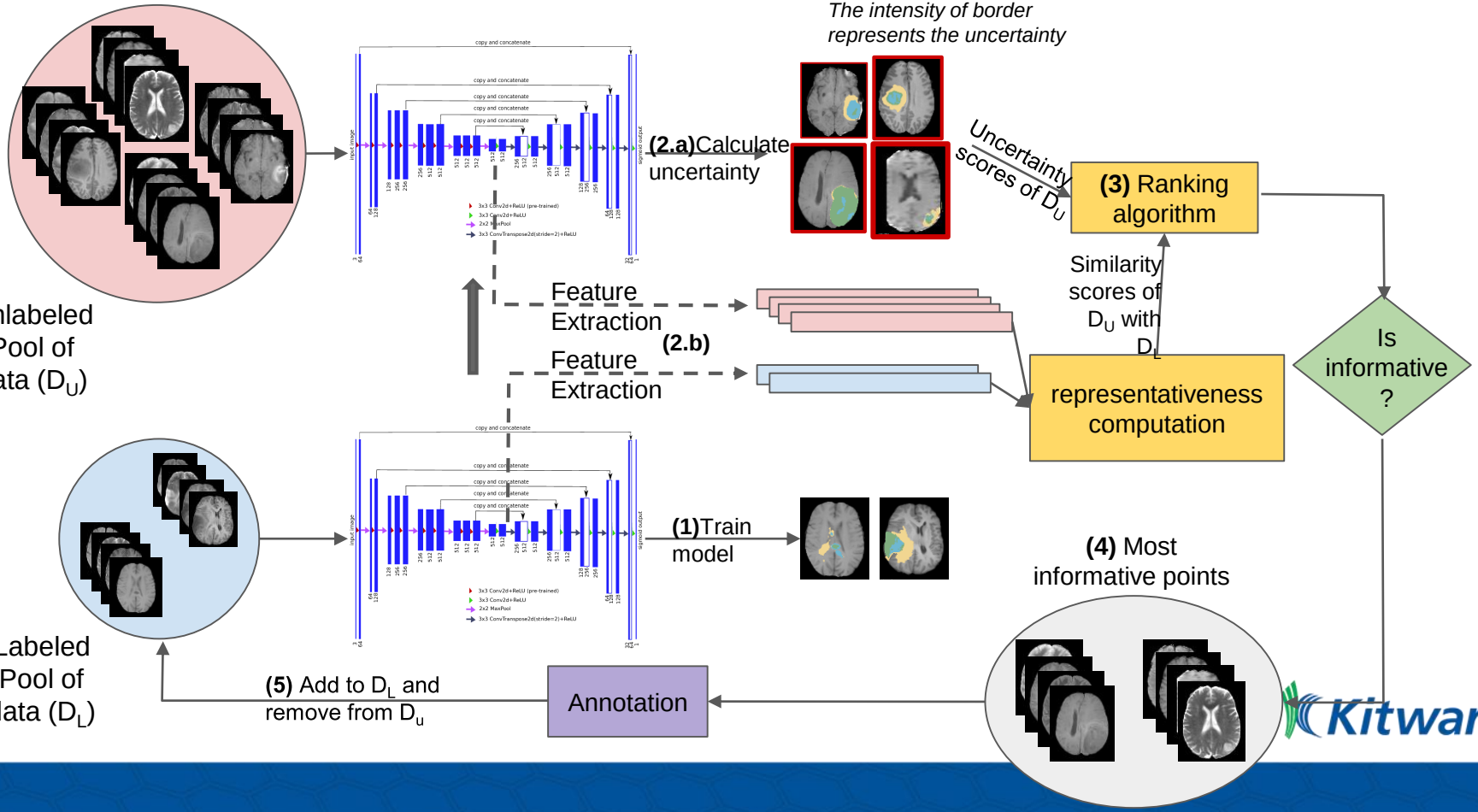


$$K(no.\,of\,clusters) = 0.8 * N + 0.2 * |D_U|$$

$$alpha = \frac{|D_{CU}|}{|D_{CU}| + |D_L U D_Q|}$$

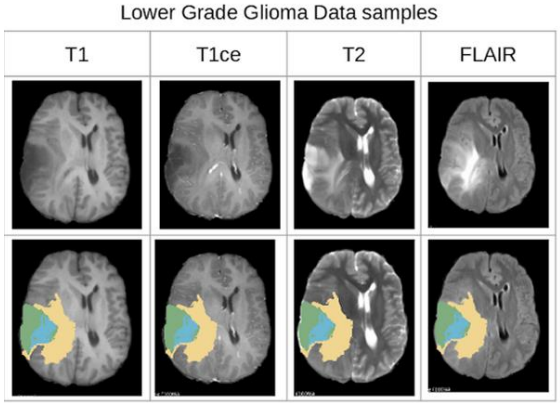
$$score = alpha * (1 - s(D_{CU}, D_L) + (1 - alpha) * U(D_{CU})$$

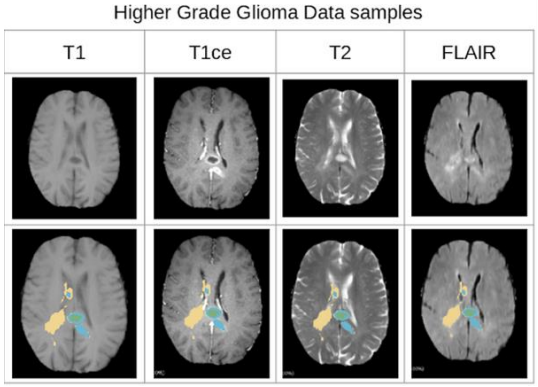
Kitware


The intensity of border represents the uncertainty
Unlabeled Pool of data (D_U)
(2.a)Calculate uncertainty
Uncertainty scores of D_U
(3) Ranking algorithm
Feature Extraction
(2.b)
Feature Extraction
Similarity scores of D_U with D_L
representativeness computation
Is informative ?
(1)Train model
(4) Most informative points
Labeled Pool of data (D_L)
(5) Add to D_L and remove from D_u
Annotation
Kitware

Data - 2018 BraTS MICCAI challenge dataset

- Consists of 210 cases of High Grade Gliomas(HGG) and 75 cases of Low Grade Gliomas(LGG)
- Each slice has been manually annotated into 4 categories - *enhancing tumor, tumor core, whole tumor*, and the *background and normal brain pixels*
- 4 modalities - T1, T1 contrast enhanced (T1ce), T2 and FLAIR





Coloring scheme - Yellow: Whole Tumor, Green: Tumor Core, Blue: Enhancing Tumor

Kitware

Preprocessing

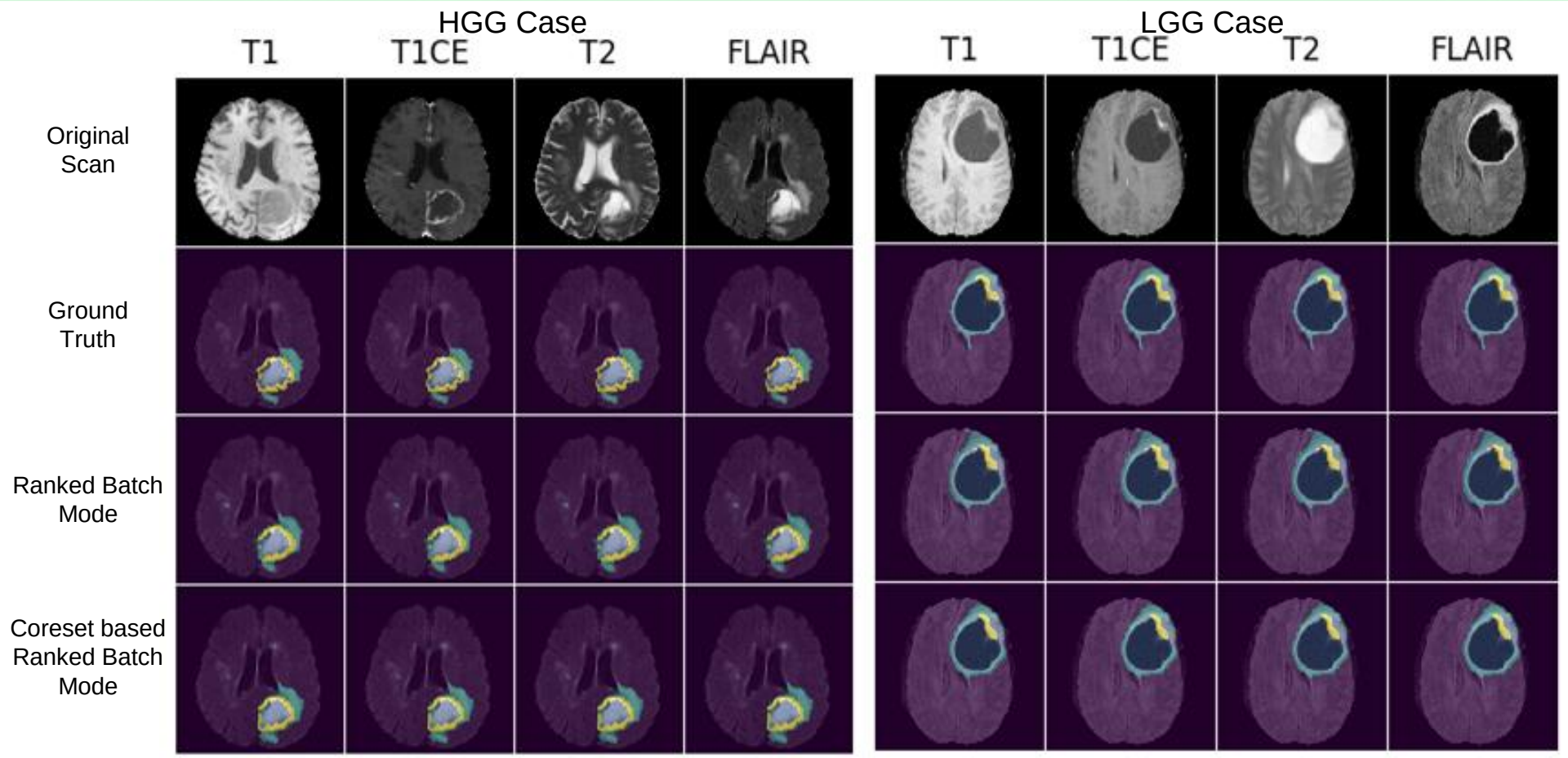
- Each slice of the four modalities for every case is **normalized** to have zero mean and unit variance
- **Patches** are **randomly sampled** from each slice after eliminating the zero-intensity pixels to tackle the class-imbalance problem
- data is randomly split into the train-validation-test parts in the ratio of **80:10:10** on case level
- This populates the t**raining data with 99,864 patches**, **validation data with 12,264 patches**, and **testing data with 12,702 patches**, each patch of size $\mathbf{128 * 128 * 4}$

Kitware

Results

Exp No.	Model	Whole Tumor Dice Coefficient	Tumor Core Dice Coefficient	Enhancing Tumor Dice Coefficient	Avg Query Time
1.	Vanilla U-Net	0.815	0.689	0.608	-
2.	U-Net with Uncertainty Sampling	0.802	0.724	0.767	-
3.	U-Net with RBM Sampling	0.829	0.812	0.788	1hr 50mins
4.	U-Net with Coreset based RBM sampling	**0.844**	**0.83**	**0.799**	**43 mins**

Kitware


HGG Case
LGG Case
T1
T1CE
T2
FLAIR
T1
T1CE
T2
FLAIR
Original Scan
Ground Truth
Ranked Batch Mode
Coreset based Ranked Batch Mode

Conclusion

- Lesser number of queries and reduced average query time
  - Faster convergence due to lesser queries
- Tackling class imbalance using Active Learning
  - Intelligently selects the under-represented class as they are more uncertain initially, and also have less representation in the labeled dataset
- *Future Work:*
  - Test against more datasets
  - Uncertainty of model using Monte Carlo Dropout

Kitware

Questions?

Kitware