Secure and Decentralized Collaboration in Oncology: A Blockchain Approach to Tumor Segmentation

Ramin Ranjbarzadeh

School of computing, Dublin City University, Ireland.
ML-Labs: SFI Centre for Research Training in Machine Learning

Introduction

1) Medical imaging is a crucial component of oncology and allowing us to observe the hidden complexities of the human body.

2) Some medical imaging techniques include MRI, CT, and PET scans.

3) Tumor segmentation is an essential step in oncological imaging.

4) The process of manual segmentation is extremely time-consuming and requires a significant amount of effort.

5) Automated segmentation techniques contribute to the standardization of evaluations across various healthcare facilities, thereby enhancing the overall quality of cancer care.

Introduction

6) The efficacy of automated segmentation systems may be constrained by the quality and amount of the data.

7) Automatic techniques in clinical workflows gives rise to concerns regarding data privacy and security.

8) Blockchain technology is a revolutionary option for addressing the crucial problems of data privacy and security.

9) This structure not only secures confidential patient data but also cultivates a sense of collaboration among healthcare practitioners.

10) By utilizing smart contracts, blockchain technology facilitates the effortless and secure exchange and annotation of medical images.

Main Objective

Introducing a new framework that uses blockchain technology to facilitate secure and decentralized communication between radiologists, doctors, clinicians, and data scientists when using real private medical data.

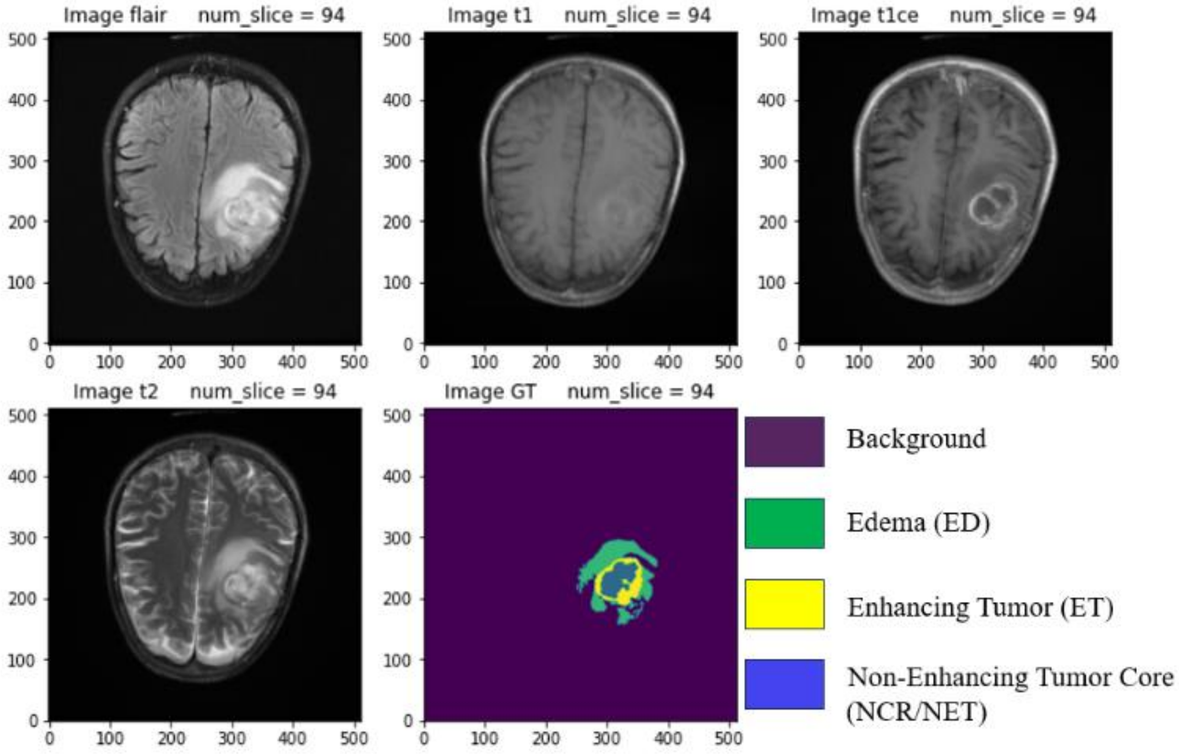

Figure 1. One sample of different brain MRI modalities from a private dataset.

Fundamentals of Blockchain Technology

1) The structure of a blockchain is comprised of a sequence of data blocks that are securely interconnected by cryptographic principles.

2) Every block consists of a set of transactions, which, in the context of medical imaging, could encompass activities like:

- Uploading a new image
- Annotating an image
- Accessing an image for analysis

3) A blockchain differs from standard databases by distributing its data over a network of computers (nodes), instead of storing all data on a central server.

4) Encryption is crucial for maintaining data security on a blockchain.

Collaborative Tumor Segmentation and Medical Imaging

1) Obtaining high-resolution medical images.

2) Encrypting medical images before uploading them to the blockchain.

3) A complex access control mechanism is implemented to annotate of these images by a specific set of medical professionals.

4) Before allowing an expert to see or make notes on an image, the smart contract validates their qualifications based on a predetermined set of standards.

5) The smart contract only allows access to the encrypted images after successful verification, enabling the authorized expert to decrypt and comment on them.

Collaborative Tumor Segmentation and Medical Imaging

6) Smart contracts simplify the pay process for experts by incentivizing and rewarding their contributions according to preset criteria.

7) Data scientists employ the dataset to enhance or create new tumor segmentation algorithms.

8) Data scientists utilize statistical and machine learning approaches to predict outcomes such as:
- Tumor progression
- Patient response to different treatment procedures
- Probability of recurrence

5) Every image, annotation, and subsequent analysis result is meticulously documented on the blockchain, forming an immutable history record.

6) The suggested system, with its focus on protecting confidential medical data is expected to gain advantages from a permissioned blockchain.

Collaborative Tumor Segmentation and Medical Imaging

1) Role-based access control (RBAC) in blockchain can be resource-intensive, particularly when it comes to gas consumption and computing costs.

2) Gas is a price paid to account for the computational energy expended.

3) To enhance these aspects, we can investigate various strategies:

- State Channels: This technique minimizes the number of transactions that must be handled on-chain by using state channels for microtransactions.
- Efficient Contract Design: Maximizing the efficiency of smart contract code by minimizing on-chain processes.
- Layered Architecture: Utilizing off-chain computations whenever feasible and reserving the blockchain just for actions that require finality and security.

```
pragma solidity ^0.8.0;

contract TumorSegmentationFramework {
    struct Image {
        string imageHash; // Hash of the encrypted image for security
        address uploader; // Address of the radiologist who uploaded the image
        bool isAnnotated; // Flag to check if the image has been annotated
        bool isProcessed; // Flag to check if the image has been processed
    }

    struct Annotation {
        string annotationData; // Encrypted data or hash of the annotation
        address[] annotators; // Addresses of experts who contributed to the annotation
        bool isValidated; // Whether the annotation has been validated
    }

    struct AnalysisResult {
        string resultData; // Encrypted data or hash of the analysis result
        address analyst; // Address of the data scientist who analyzed the image
    }

    mapping(uint => Image) public images; // Mapping of image IDs to Image structs
    mapping(uint => Annotation) public annotations; // Mapping of image IDs to Annotation structs
    mapping(uint => AnalysisResult) public analysisResults; // Mapping of image IDs to AnalysisResult structs

    // 1. Image Obtaining
    function uploadImage(string memory _imageHash) public {
        // Implementation for uploading an encrypted image to the blockchain
        // Set the isAnnotated and isProcessed flags to false initially
    }

    // 2. Image Annotating
    function annotateImage(uint _imageId, string memory _annotationData, address _annotator) public {
        // Implementation for experts to annotate an image
        // Include validation checks to ensure only authorized experts can annotate
        // Update the isAnnotated flag once the annotation is complete and validated
    }

    // Validation of Annotations
    function validateAnnotation(uint _imageId) private {
        // Implementation for validating the annotation
        // This could involve consensus mechanisms among experts
        // Update the isValidated flag upon successful validation
    }

    // 3. Image Processing
    function processImage(uint _imageId, string memory _resultData, address _analyst) public {
        // Implementation for data scientists to submit their analysis of the annotated images
        // Ensure the image has been annotated and validated before processing
        // Update the isProcessed flag once analysis is complete
    }

    // Utility Functions (e.g., access control, consensus mechanism, compensation)
    // Include functions for controlling access based on roles, building consensus on annotations,
    // and compensating contributors based on predefined criteria.
```

Thank You