

Sprint 1 : Cervical spine fracture detection

A1_10

Intro

- Over 1.5 million spine fractures occur annually
- Analyzing medical images can often be a difficult and time consuming process

--->It is increasingly urgent to use artificial intelligence instead of human force for Cervical Spine Fracture Detection

Product Mission

- Developing a computer-aided diagnosis software
- Quickly detecting the location of any vertebral fractures
- Improving the accuracy of detection of bone fractures
- Providing a user-friendly interface for radiologists to use

Unlike identifying manually, our product is totally **automatic** and **easy to use** with **human-comparable accuracy**

Users: Radiologists

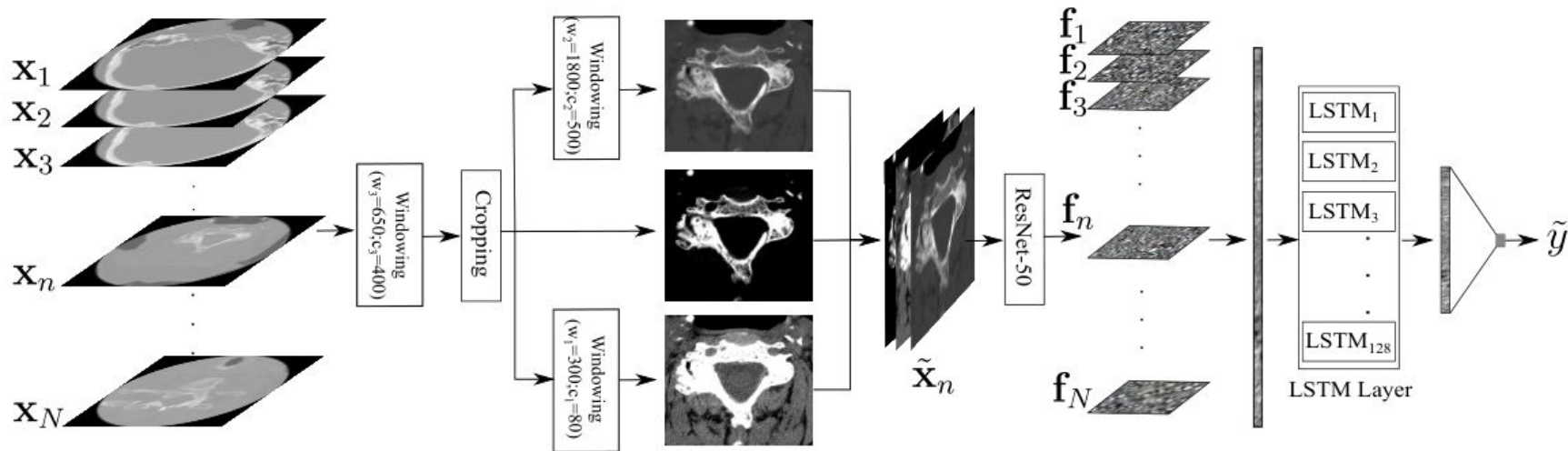


User Stories

As a radiologist, I want a computer software to help me detect spine fractures so that we can increase the work efficiency and reduce the possibility of undetected fractures.

Comprehensive literature review

- Preprocessing—Training model



- The preprocessing part: each CT image is duplicated to a set of three images, and then the images are cropped and centered to the middle of spine bones. All of the processed images are resized to a resolution of 384x384.
- The training part: Resnet-50 is used to extract the feature maps

Bidirectional Network of LSTM (BLSTM) to generate label

Minimum Valuable Product

- An algorithm that can automatically detect cervical spine fractures with reasonable resources in several minutes
- Can provide accurate fracture segmentation with little or no manual modification

Development environment

- Pycharm
 - To develop and debug program on local machines
- Github
 - To cooperate and control the code version
- Anaconda
 - For better python environment management
- Cuda
 - For faster training speed by GPU computing

Technologies

- Pytorch
 - Popular deep learning framework, open source and with community support
- Pydicom
 - To process dicom tag files
- U-Net CNN
 - Widely used in medical image segmentation
- Resnet
 - Some successful examples to review
- PyQt5
 - Widely used GUI toolkit
 - Cross-platform
- AWS, Google Cloud Platform



Next Sprint

- Download the kaggle dataset in dicom format and view the data
- Process the dicom file with pydicom module
 - Data Preprocessing (Normalization, Windowing, Centering)
- Implement a simple CNN model with pytorch
 - Coding and debugging
- Train with a small size dataset to test our model

Thanks!