Highdicom – High-level DICOM Abstractions for the Python Programming Language to Encode Image-derived Annotations and Machine Learning Outputs in Standard Format

Markus D. Herrmann, MD, PhD, Massachusetts General Hospital and Harvard Medical School; Andriy Y. Fedorov, PhD; Steve Pieper, PhD; Sean W. Doyle; Christopher P. Bridge, PhD; David A. Clunie, MBBS; Jochen K. Lennerz, MD, PhD

Background/Problem Being Solved

Artificial intelligence and machine learning (ML) promise to revolutionize digital pathology and radiology and unlock the full potential of image-based diagnostics. Several ML models have shown promising results in various research settings. Pathologists and radiologists are now tasked with assessing their performance in clinical practice. A major barrier for integrating and assessing models is the lack of interoperable interfaces between ML models and clinical systems.
The DICOM standard specifies information object definitions and services for representation and communication of images and image-related information, including annotations and analysis results. However, most ML model developers are not leveraging the DICOM standard for encoding and exchanging model inputs and outputs and instead rely on custom formats that impede interoperability.

Interventions

We created a software library called highdicom, which implements DICOM service-object pair (SOP) classes in Python and exposes an object-orientated interface that enables ML model developers to create image-derived DICOM segmentation and structured report (SR) objects without having to understand lower level details of the standard.

Barriers/Challenges

The DICOM standard comprises an extensive set of documents that specify many different aspects of medical imaging and provides numerous IODs for a wide variety of imaging modalities and use cases. The comprehensiveness and complexity of the standard can be daunting, especially to data scientists who are new to the field of medical imaging informatics. Implementing the standard correctly can be challenging even for experienced DICOM software developers. Furthermore, while well-established libraries exist for creating DICOM objects for the C^{++} and Java programming languages, ML models are typically developed in Python but high-level Python abstractions for creating image-derived DICOM objects are currently missing.

Outcome

Highdicom exposes intuitive interfaces that abstract the complexity of the DICOM standard. The library facilitates the construction of DICOM instances for representation of image regions of interest (ROIs) as well as image-derived annotations, measurements, and qualitative evaluations. Specifically, it enables the creation of DICOM segmentations derived from various types of images, including large 2D tiled whole slide microscopy images in pathology as well as 3D computed tomography and magnetic resonance images in radiology. In addition, highdicom enables the creation of DICOM SR instances containing image-derived ROI definitions, measurements or annotations with few lines of Python code. Furthermore, highdicom classes perform automatic DICOM validation upon object construction and ensure that created data sets are compliant with the standard and interoperable with other medical imaging software.

Conclusion/Statement of Impact

The presented library enables ML model and application developers to encode inputs and outputs in standard format and facilitate clinical integration and validation by achieving interoperability with enterprise imaging systems. We made highdicom available as free and open source software in the hope that the library helps promoting data standardization in ML research and streamline ML model development and clinical integration to ultimately drive innovation for patients.

Lessons Learned

The DICOM standard is vast and complex and at times appears unnecessarily complicated. However, requirements and constraints specified by the standard are invaluable for clinical integration of machine learning models. The complexity of DICOM can be significantly abstracted via libraries and tools without compromising interoperability.

Keywords

machine learning, data standardization, DICOM, Python, radiomics, computational pathology