Supplementary

PyRadiomics

PyRadiomics is a Python-based open-source package designed for extracting radiomics features from medical imaging data. The official website link is https://pyradiomics.readthedocs.io.

Contents of the params.yaml file


```
setting:
 binWidth: 25
 label: 1
 interpolator:'sitkBSpline' # This is an enumerated value; None is not allowed
 resampledPixelSpacing: # This disables resampling, as it is interpreted as None; to enable it, specify spacing in x, y, z as [x, y, z]
 weightingNorm: # If no value is specified, it is interpreted as None

# Image types to use: "Original" for unfiltered image; for possible filters, see documentation.
# Some feature extraction requirements are added here
imageType:
 Original: {} # for dictionaries/mappings, None values are not allowed, '{}' is interpreted as an empty dictionary
 LoG: {}
Wavelet: {}
 Square: {}
 SquareRoot: {}
 Logarithm: {}
Exponential: {}
 Gradient: {}
LBP3D: {}

# Feature classes, from which features must be calculated. If a feature class is not mentioned, no features are calculated for that class.
Otherwise, the specified features are calculated, or if none are specified, all are calculated (excluding redundant/deprecated features).
featureClass:
 # redundant Compactness 1, Compactness 2 and Spherical Disproportion features are disabled by default, they can be enabled by
specifying individual feature names (as is done for glcm) and including them in the list.
 shape:
 shape2D:
 firstorder: [] # specifying an empty list has the same effect as specifying nothing.
 glcm:  # Disable SumAverage by specifying all other GLCM features available
 glrlm: # for lists, None values are allowed; in this case, all features are enabled
 glszm:
 gldm: # contains deprecated features, but as no individual features are specified, the deprecated features are not enabled
 ngtdm:
```


Feature classes and filters

1. First Order Features ----First-order statistics describe the distribution of voxel intensities within the image region defined by the mask through commonly used and basic metrics.

2. Shape Features (3D)-----In this group of features, we included descriptors of the three-dimensional size and shape of the ROI. These features are independent to the gray level intensity distribution in the ROI and are therefore only calculated on the nonderived image and mask.

3. Gray Level Co-occurrence Matrix (GLCM) Features-----A gray level co-occurrence matrix (GLCM) of size $N_g \times N_g$

https://dx.doi.org/10.21037/qims-22-848

describes the second-order joint probability function of an image region constrained by the mask and is defined as P(i,j|δ,θ).

4. Gray Level Size Zone Matrix (GLSZM) Features-----A gray level size zone matrix (GLSZM) quantifies gray level zones in an image. A gray level zone is defined as the number of connected voxels that share the same gray level intensity.

5. Gray Level Run Length Matrix (GLRLM) Features-----A gray level run length matrix (GLRLM) quantifies gray level runs, which are defined as the length, in number of pixels, of consecutive pixels that have the same gray level value.

6. Neighboring Gray Tone Difference Matrix (NGTDM) Features—A neighboring gray tone difference matrix quantifies the difference between a gray value and the average gray value of its neighbors within distance δ.

7. Gray Level Dependence Matrix (GLDM) Features-----A gray level dependence matrix (GLDM) quantifies gray level dependencies in an image. A gray level dependency is defined as the number of connected voxels within distance δ that are dependent on the center voxel.

Aside from the feature classes, there are also some built-in optional filters:

1. Laplacian of Gaussian (LoG, based on SimpleITK functionality)-----Applies a Laplacian of Gaussian filter to the input image and yields a derived image for each sigma value specified. A Laplacian of Gaussian image is obtained by convolving the image with the second derivative (Laplacian) of a Gaussian kernel.

2. Wavelet (using the PyWavelets package)-----Applies wavelet filter to the input image and yields the decompositions and the approximation.

3. Square-----Computes the square of the image intensities.

4. Square Root----- Computes the square root of the absolute value of image intensities.

5. Logarithm----- Computes the logarithm of the absolute value of the original image + 1.

6. Exponential----- Computes the exponential of the original image.

7. Gradient----- Computes and returns the gradient magnitude in the image.

8. Local Binary Pattern (3D)----- Computes and returns the local binary pattern (LBP) in 3D using spherical harmonics.

The features extracted in the study are as follows:

14 original_shape features; 18 original_firstorder features; 24 original_glcm features; 16 original_glrlm features; 16 original_glszm features; 14 original_gldm features; 5 original_ngtdm features; 744 wavelet and subclass filters features; 93 square filters features; 93 squareroot filters features; 93 logarithm filters features; 93 exponential filters features; 93 gradient filters features; 279 lbp-3D and subclass filters features; and 5 diagnostics_Image/Mask features.

https://dx.doi.org/10.21037/qims-22-848

Table 1 Results of the tenfold cross-validation experiment

	Precision	Recall	F1	Acc
First experiment				
Training cohort	0.98	0.98	0.98	0.99
Testing cohort	0.84	0.89	0.84	0.84
Second experiment				
Training cohort	0.95	0.93	0.98	0.93
Testing cohort	0.87	0.85	0.84	0.85
Third experiment				
Training cohort	0.98	0.98	0.98	0.98
Testing cohort	0.93	0.93	0.92	0.93
Fourth experiment				
Training cohort	0.99	0.99	0.99	0.99
Testing cohort	0.91	0.89	0.89	0.89
Fifth experiment				
Training cohort	0.99	0.99	0.99	0.99
Testing cohort	0.91	0.90	0.91	0.91
Sixth experiment				
Training cohort	0.99	0.99	0.99	0.99
Testing cohort	0.98	0.98	0.98	0.98
Seventh experiment				
Training cohort	0.99	0.99	0.99	0.99
Testing cohort	0.90	0.88	0.88	0.88
Eighth experiment				
Training cohort	0.98	0.97	0.97	0.97
Testing cohort	0.90	0.89	0.89	0.89
Ninth experiment				
Training cohort	0.99	0.99	0.98	0.99
Testing cohort	0.95	0.95	0.96	0.95
Tenth experiment				
Training cohort	0.99	0.99	0.99	0.99
Testing cohort	0.93	0.92	0.92	0.92

The dataset was split into 10 equal parts. Two parts were randomly selected as the testing cohort each time, and the remaining 8 were the training cohort.

 https://dx.doi.org/10.21037/qims-22-848

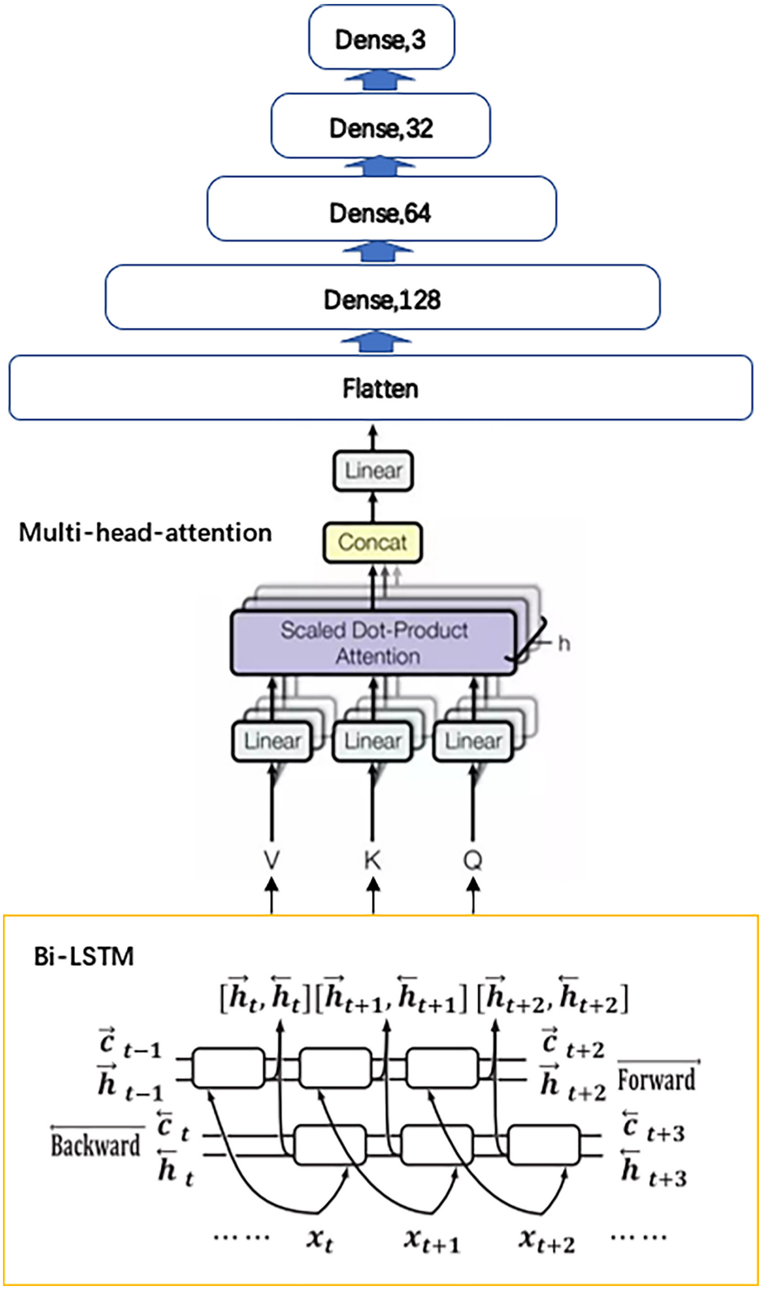

Figure S1 Detailed structure of the model in this study.

https://dx.doi.org/10.21037/qims-22-848

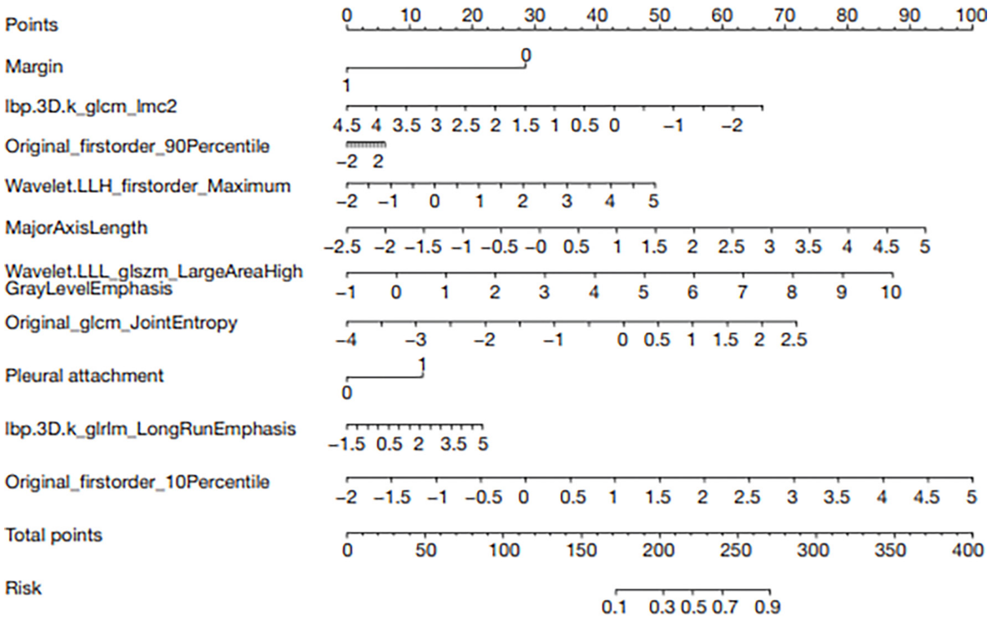

Figure S2 The nomogram for predicting IAC risk was constructed based on 3D radiomic signatures and radiological features. IAC, invasive adenocarcinoma.

https://dx.doi.org/10.21037/qims-22-848

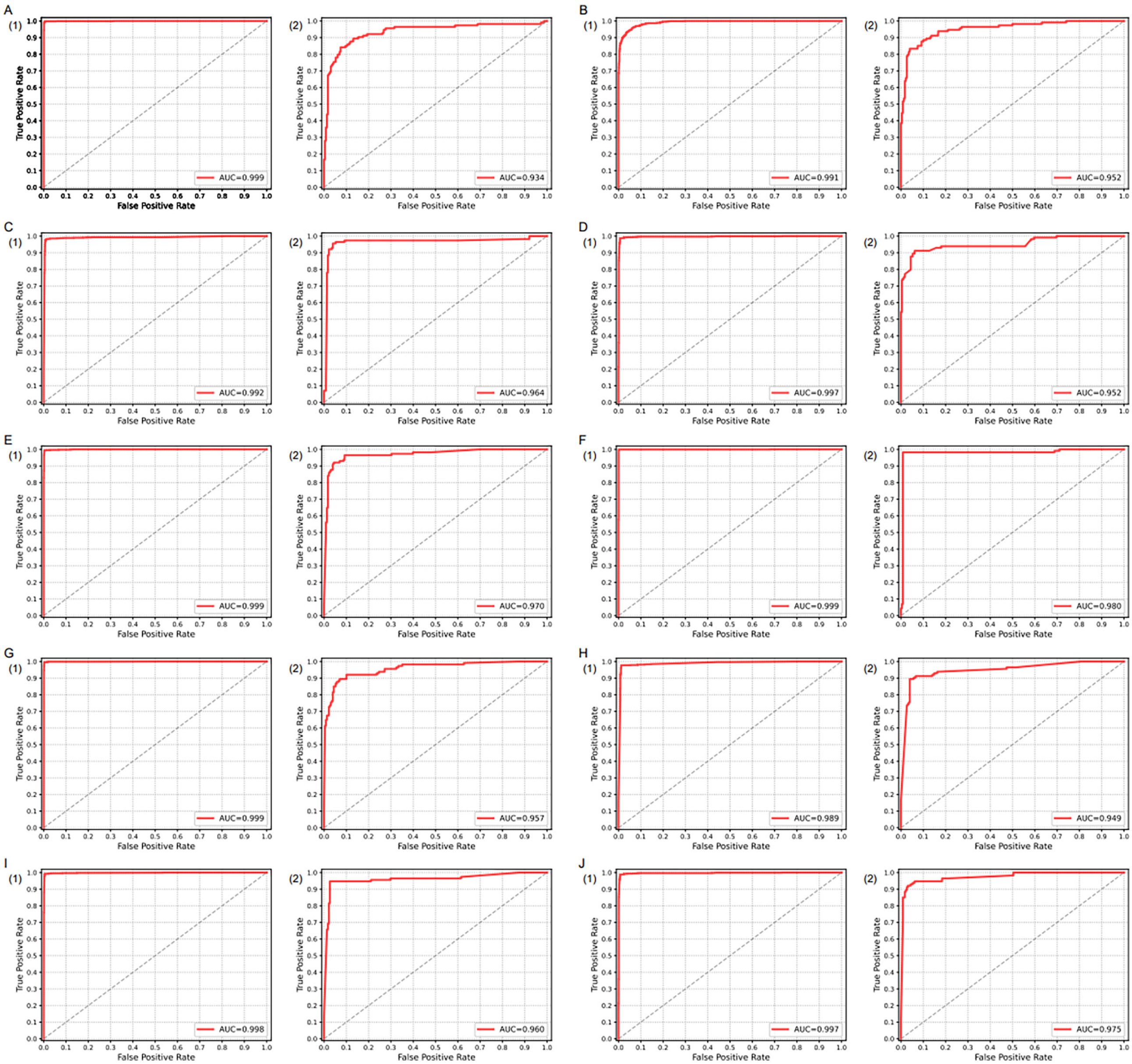

Figure S3 The ROC curves of the training set and test set in each tenfold cross-validation experiment. Subgraph (1) in each group represents the ROC curve of the training cohort, and subgraph (2) represents the ROC curve of the testing cohort.

 https://dx.doi.org/10.21037/qims-22-848

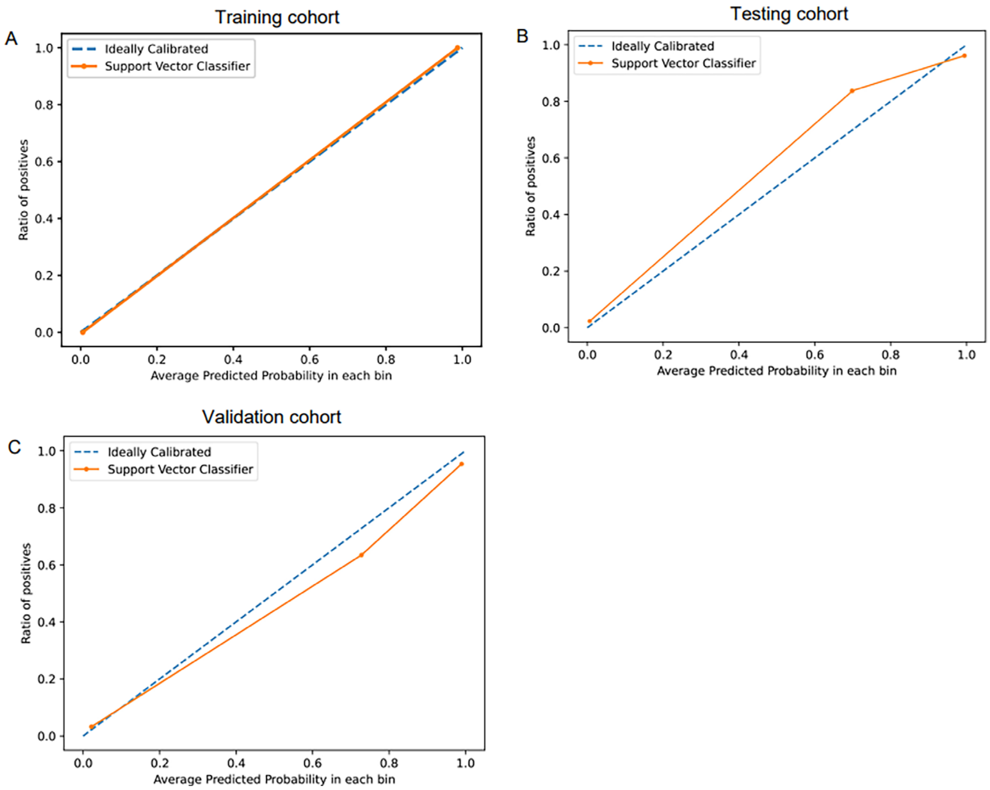

Figure S4 Calibration plot of the model in three data cohorts. A represents the calibration plot in the training cohort. B represents the calibration plot in the testing cohort. C represents the calibration plot in the verification cohort.

https://dx.doi.org/10.21037/qims-22-848

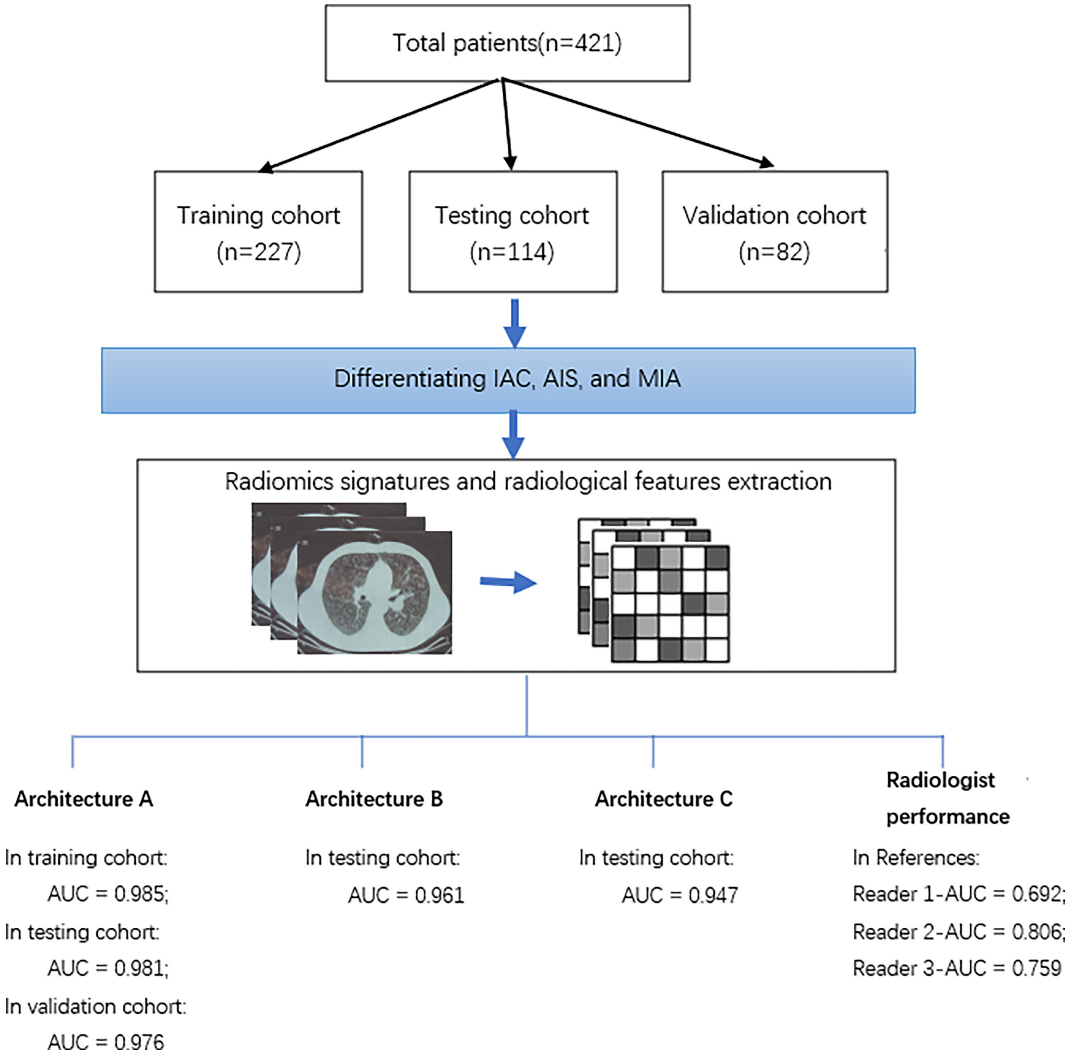

Figure S5 Overall study design flowchart for the training, testing, and validation cohorts and diagnostic performance by each approach.

https://dx.doi.org/10.21037/qims-22-848