Deep Learning for Pulmonary Embolism Detection: Tackling the RSNA 2020 AI Challenge

Ian Pan, MD¹

From the University Hospitals Cleveland Medical Center, 11100 Euclid Ave, Cleveland, OH and Case Western Reserve University School of Medicine, Cleveland, Ohio. Received March 1, 2021; revision requested April 12; revision received May 18; accepted May 24. Address correspondence to the author (e-mail: ianpan358@gmail.com).

¹Current address: Department of Radiology, Brigham and Women's Hospital, Harvard Medical School, Boston, Mass

Authors declared no funding for this work.

Conflicts of interest are listed at the end of this article.

Radiology: Artificial Intelligence 2021; 3(5):e210068 • <https://doi.org/10.1148/ryai.2021210068> • Content codes: **AI** **CH**

In 2020, the Radiological Society of North America and Society of Thoracic Radiology sponsored a machine learning competition to detect and classify pulmonary embolism (PE). This challenge was one of the largest of its kind, with more than 9000 CT pulmonary angiography examinations comprising almost 1.8 million expertly annotated images. More than 700 international teams competed to predict the presence of PE on individual axial images, the overall presence of PE in the CT examination (with chronicity and laterality), and the ratio of right ventricular size to left ventricular size. This article presents a detailed overview of the second-place solution. Source code and models are available at <https://github.com/i-pan/kaggle-rsna-pe>.

© RSNA, 2021

In 2020, the Radiological Society of North America (RSNA) hosted its fourth machine learning (ML) competition together with the Society of Thoracic Radiology (STR), the RSNA-STR Pulmonary Embolism Detection challenge (1). This challenge included 9446 CT pulmonary angiographic (CTPA) examinations, expertly annotated by thoracic radiologists from the STR. The data were divided into training ($n = 7279$), public test ($n = 650$), and private test ($n = 1517$) sets, and the final standings were determined based on private test set performance. Throughout the competition, participants were allowed to evaluate their model up to five times per day on the public test set. These results were made available on a public leaderboard, which allowed one to compare a model's performance against others. However, private test set results were not revealed until the competition ended. Because each CTPA examination contained up to several hundred images, a total of 1 790 595 individually annotated images were available for training. Importantly, the format of this competition was a code competition: Participants had to submit inference code that ran on the test data, as opposed to simply submitting predictions. This format encouraged users to consider computational constraints when building their models because it enforces a strict runtime limit and also somewhat levels the playing field because all models run inference on the same computational resources. It also allows for the private test set to be completely hidden and adds to the integrity of the competition results.

Understanding the Challenge

A robust understanding of the task at hand is critical to any ML problem, and even more so for competitions where the rules and objectives are set in place by an external body. Given a supervised ML problem, the first step should be to define the prediction task. In this competition, 10 classification labels were used as part of the scoring and are summarized in Table 1. These labels indicated the follow-

ing: whether pulmonary embolism (PE) was present on a specific image; whether PE was present in a single examination, or if the study was indeterminate; location of PE (left-sided, right-sided, and/or central); acuity of PE (acute and/or chronic); and whether the ratio of right ventricular size to left ventricular size (RV/LV) was greater than or equal to 1, suggestive of right heart strain. Only one classification label (presence of PE) was at the level of a single image; all others were at the examination level. This task, then, was a multilabel binary classification problem, which is analogous to multiple true/false questions on an examination.

One of the most important steps in approaching a ML problem is establishing the metric by which performance will be evaluated. In the real world, the best metric is one that most accurately captures the goal of the model (eg, high sensitivity and negative predictive value for a population screening model or high specificity and positive predictive value for a confirmatory diagnostic model). Multiple metrics often are reported to obtain a more holistic understanding of the model's performance; however, for competitions, the host sets the metric, and only a single metric, which can be a combination of multiple metrics, determines the winner.

In this competition, the metric was a weighted log loss. Log loss is equivalent to binary cross-entropy, commonly used to train ML models. The first component of the metric was an image-level weighted log loss, calculated as the simple log loss for each image multiplied by the proportion of positive images in the examination. For example, if the loss of image #25 of examination #408 was 0.5 and 20% of the examination's images were positive for PE, the loss of that image is equal to $0.2 \times 0.5 = 0.1$. The loss is then averaged across all images. A crucial aspect of this loss is that images belonging to examinations without PE will not contribute to the loss because the proportion of positive images would be 0. The other components of the log loss were the simple

Abbreviations

AUC = area under the receiver operating characteristic curve, CNN = convolutional neural network, CTPA = CT pulmonary angiography, CV-AUC = cross-validation AUC, ML = machine learning, PE = pulmonary embolism, RSNA = Radiological Society of North America, RV/LV = right ventricular size to left ventricular size, STR = Society of Thoracic Radiology, 3D = three dimensional

Summary

Machine learning competitions promote the development of state-of-the-art models for various tasks. This work outlines the thought process behind the second-place solution to the 2020 RSNA-STR Pulmonary Embolism Detection challenge.

Keywords

CT, Neural Networks, Thorax, Pulmonary Arteries, Embolism/Thrombosis, Supervised Learning, Convolutional Neural Networks (CNN), Machine Learning Algorithms

Table 1: Classification Labels and Associated Weights Involved in Scoring for the RSNA-STR Pulmonary Embolism Detection Challenge

Label	Description	Weight
C ₀	Image-level presence of PE*	0.0736
C ₁	Negative examination for PE	0.0736
C ₂	Indeterminate	0.0920
C ₃	Chronic PE	0.1043
C ₄	Acute and chronic PE	0.1043
C ₅	Central PE	0.1877
C ₆	Left-sided PE	0.0626
C ₇	Right-sided PE	0.0626
C ₈	RV/LV ratio ≥ 1	0.2347
C ₉	RV/LV ratio < 1	0.0782

Note.—PE = pulmonary embolism, RSNA = Radiological Society of North America, RV/LV ratio = right ventricular size-to-left ventricular size ratio, STR = Society of Thoracic Radiology. *Denotes image-level label. All other labels are examination-level.

log loss of the examination labels. Figure 1 explains how the loss is computed as well as the different weights of the components.

There also was a label consistency requirement to enforce “sensible” model predictions, detailed on the Kaggle website (2). In essence, this requirement enforced the idea that if an examination was predicted to be positive ($P > .5$), then certain other labels should also have a positive prediction ($P > .5$). If an examination was negative, certain other labels should not have a positive prediction.

After thoroughly dissecting the competition task, metric, and rules, model development can begin with a clear objective in mind.

Understanding the Data

Prior experience with medical imaging was advantageous because familiarity with the Digital Imaging and Communications in Medicine (DICOM) format and the specifics of CT imaging allowed for better modeling decisions. Images in a CT examination were sorted sequentially according to image position

coordinates available in the DICOM header, which was key to a competitive solution. In addition, understanding CT attenuation values and how images are windowed into 8-bit pixel values (from 0 to 255) was also crucial. Last year’s RSNA Intracranial Hemorrhage Detection challenge (3) demonstrated that (a) incorporating neighboring section information significantly improved performance and (b) fixed windowing, as opposed to a learnable windowing layer as described in Lee et al (4), was sufficient for a strong solution. A fixed PE-specific window of level 100 HU and width 700 HU taken from Wittram et al (5) was used for my solution. Also, given that the finding of PE on a section of an examination is often small relative to the overall image, each image was maintained at its original resolution of 512×512 pixels. In addition, given that in a CT section much of the periphery is empty, images were randomly cropped to 448×448 pixels during training for additional data augmentation and center-cropped to 448×448 pixels during inference (Fig 2). Larger and more complex models required a reduction in image resolution, specified in the relevant sections.

Data augmentation is key to building a strong model. The RandAugment strategy was used due to its simplicity and efficacy (6). Transforms applied prior to windowing included spatial transforms (shear, translation, rotation, zoom, flip) and random CT attenuation shift (randomly adding a fixed attenuation to all pixels in the image), although applying transforms after windowing would likely have achieved similar results.

Setting Up Local Validation

A robust local validation setup is extremely important as it allows one to rapidly iterate without having to rely on public test set submissions. A common pitfall in ML competitions is overreliance on public test set results to inform model development. Eventually, as public test set performance increases, the model begins to overfit to noise and patterns in the public test set. Thus, a robust local validation setup serves as an additional barrier to overfitting by providing another gauge as to whether improvements in model performance are real or spurious. In general, correlations between the trends of the local validation and public test set performance are most helpful (ie, a modification to the model increases/decreases both local validation and public test set scores, even if the scores themselves are different). Incongruent performance changes (ie, increase in local validation and decrease in public test set or vice versa) should prompt consideration of whether the corresponding modification is truly beneficial or whether there are systematic differences between the training and test data.

My local validation setup was simple fivefold cross-validation: Each experiment was repeated five times using different training (80% of available data) and validation (20% of available data) sets, allowing for better performance estimates. Because competition success relies on heavy experimentation, it can be easy to overfit to a single validation fold. It can be advantageous to have an additional holdout test set outside of the cross-validation to test ensembles or determine thresholds for prediction. In this case, I did not feel it was necessary to reserve an additional holdout set.

Ultimately, performance on the private test set determines the final standing. A robust local validation is useful insofar

$$LL = -[y \log p + (1 - y) \log (1 - p)]$$

1. Calculate log loss (LL) for each image in the dataset, where p is the prediction and y is the ground truth (presence of PE on that image).
2. For each image M_i belonging to exam E_j , assign a weight equal to $W_0 * q_j$, where q_j is the proportion of PE-positive images in exam E_j .
3. For each exam E_j and exam-level label C_k , calculate the log loss and assign a weight W_k , where k is an integer from 1 to 9 (see right).
4. Compile all of the calculated losses row-wise, such that each loss corresponds to a single row. The number of rows is equal to (number of images) + [(number of exams) x (number of exam-level labels)].
5. Multiply each loss by its corresponding weight, as defined above.
6. Average the weighted loss across all rows.
7. Average the weights across all rows. Many image-level weights will be 0, corresponding to images from exams without PE.
8. Divide the average weighted loss (6) by the average weight (7).

$W_0 = 0.0736$
 $W_1 = 0.0736$
 $W_2 = 0.0920$
 $W_3 = 0.1043$
 $W_4 = 0.1043$
 $W_5 = 0.1877$
 $W_6 = 0.0626$
 $W_7 = 0.0626$
 $W_8 = 0.2347$
 $W_9 = 0.0782$

Figure 1: Explanation of how to calculate the weighted log loss metric for the challenge. Each class label is represented by the shorthand in Table 1.

Figure 2: Cropping scheme used during model training. Top: Images were randomly cropped during training for additional data augmentation. Bottom: Images were center-cropped during inference. Larger models adopted the same cropping scheme with downscaling of the image resolution.

as it is able to reflect performance on the private test set. For example, during the RSNA Pneumonia Detection challenge, local validation was not as useful due to significant differences in the labeling of training and test data (7). My intuition was that this competition was more akin to last year's intracranial hemorrhage challenge in which local validation, public leaderboard, and private leaderboard scores were highly correlated. Although these considerations may seem exclusive to competitions, they do bear relevance to real-world model development. It is important to consider whether the validation and test sets used for development and evaluation will reflect real-world performance and what steps, if any, can be taken to improve the model's robustness to unseen data from unknown distributions.

Development Environment

Models were trained using PyTorch 1.6 (<https://www.pytorch.org>) and Python 3.7 (Python Software Foundation, <https://www.python.org>) across four NVIDIA Quadro RTX 6000 GPUs, each with 24-GB VRAM. PyTorch 1.6 and onward offers native automatic mixed precision training (8), which significantly reduces memory usage during training.

Recording Experiments

Rapid and efficient experimentation is essential to success. Given the limited time frame and relative paucity of computational resources, extensive hyperparameter tuning and large grid or randomized searches are not feasible. On the basis of experience with

prior ML competitions, I developed a modular experimentation pipeline (example available at: <https://github.com/i-pan/kaggle-rsna-pe>), which allows for rapid implementation of new ideas without extensive revision of existing code. Each experiment is represented by a configuration file, permitting rapid and simple review of an experiment's parameters to determine which modifications were helpful. This is also advantageous for reproducibility; each configuration file contains all the necessary parameters to reproduce an experiment's results.

Both the validation loss and the area under the receiver operating characteristic curve (AUC) were computed for each individual binary label. The AUC is a more interpretable measure of a model's discriminatory capabilities, allowing for a better assessment of the difficulty of each classification subtask. In general, for a given label, loss and AUC were inversely correlated (ie, lower loss was associated with higher AUC).

Building the Model

The three-dimensional (3D) nature of the data provided a rich playground for model development. Top solutions from last year's RSNA challenge, which focused on intracranial hemorrhage classification on head CT images, leveraged sequence modeling to refine image-level predictions and achieve state-of-the-art results and formed the basis for my solution. Given the relatively short 6-week time frame of the competition, it was important to pursue proven high-yield solutions.

In summary, my solution involved the following:

1. Image-level feature extraction using a convolutional neural network (CNN);
2. Sequence modeling for both refinement of image-level PE predictions and prediction of examination-level PE labels;
3. Time-distributed CNN, comprising a feature extractor and sequence model head, trained end-to-end to predict examination-level PE labels;
4. Identification of images within each examination that contained a portion of the heart;
5. RV/LV ratio feature extraction using 3D CNN, with input filtered by the heart classification model in 4; and
6. Refinement of RV/LV ratio predictions with a simple linear layer, taking in as input features extracted by the model in 5 and examination-level PE predictions by the models in 2 and 3.

Figure 3 illustrates the final integrated inference pipeline. The final ensemble consisted of two feature extractors, 12 transformers, five time-distributed CNNs, one heart section classifier, five RV/LV ratio 3D feature extractors, and five RV/LV ratio linear models and took approximately 7–8 hours to run through both public and private test data. Time-distributed and 3D CNNs required the most computing resources during training due to high memory requirements, and training the feature extractor was also computationally intensive due to the large number of images in the dataset. Models in 1, 2, and 3 bear the most resemblance to top solutions from the intracranial hemorrhage classification challenge, whereas the other models were unique to this challenge. In this competition, the metric was equivalent to the binary cross-entropy loss, which essentially eliminated the need

to experiment with different loss functions. All models were trained with a binary cross-entropy loss with weights as outlined above, with the exception of the image-level feature extraction models where labels were equally weighted during training.

Image-Level Feature Extraction

This approach relied on a strong feature extractor because the features would ultimately be fed into a sequence model for both image-level and examination-level predictions. Training a supervised feature extractor is simple: train a CNN on the ground truth labels, then remove the final linear layer that maps features to predictions. The ResNeSt-50 CNN architecture was used as the feature extractor. At the time of writing, I predominantly used EfficientNet and ResNeSt architectures due to their strong performance on ImageNet and other tasks (9,10). Experiments with EfficientNet-B4 and ResNeSt-50 demonstrated a small improvement with ResNeSt-50. For competitions, I generally evaluate moderate-to-large CNNs (eg, EfficientNet-B4 through B7, ResNeSt50 through 200) to determine whether increased model capacity results in higher performance and whether the increase in performance justifies the longer training and inference times. Given the size of this dataset and the code competition constraints, experiments were conducted only with moderate-size CNNs, using model architectures and pretrained weights from the PyTorch Image Models library (11).

The ResNeSt-50 feature extractor was trained on individual sections. The input to the network was a three-channel image comprising the image of interest and its two neighboring images (one position above and below). This allowed a two-dimensional network to take neighboring section information into context without 3D convolutions and also permitted the use of ImageNet-pretrained CNNs. ImageNet-pretrained weights generally converge faster and require less hyperparameter tuning than training from randomly initialized weights. Generalized mean pooling replaced global average pooling at the end of the network because this has been shown to improve the quality of extracted features (12). An additional linear layer was applied to reduce the feature dimensionality from 2048 to 512 to decrease the number of parameters for the subsequent sequence model.

The feature extractor was trained to predict one image-level label (presence of PE) and six examination-level labels. Image-level labels for six examination-level labels were obtained by simply transferring the examination-level label to the positive sections in that examination. Although this obviously results in mislabeled sections, the goal of building the feature extractor is to create a network that can extract rich features. Predicting multiple labels, even if noisy, would likely result in better feature extraction. The fivefold single-model cross-validation AUC average (CV-AUC) for image-level PE classification across negative and positive examinations was 0.959 (range, 0.955–0.963).

Sequence Modeling

Early sequence modeling focused on natural language processing using recurrent neural networks and modifications thereof, such as long short-term memory and gated recurrent units (13,14). Since then, significant advancements have been made, most notably the transformer architecture, which has become

Figure 3: Schematic outlines the full inference pipeline.

the new state-of-the-art for natural language processing (15). Recently, transformers have been adapted for image classification and object detection (16,17). Thus, transformers were selected, with implementations from the Transformers library, for sequence modeling due to their excellent performance and ease of training (18).

Transformers, and sequence models in general, can be adapted to predict labels for each element of the sequence or the entire sequence, which was ideal for this competition. Two separate transformers were trained for image-level and examination-level PE predictions.

After the feature extractor was trained, features (one-dimensional vector of size 512) were extracted from all images and saved to disk. Transformers can accommodate inputs of varying lengths; however, during training, sequences were padded or truncated to length 512 to accommodate minibatch training. During inference, the original unmodified sequence was fed as input.

As noted above, only images from examinations positive for PE contributed to the image-level loss. Thus, it was advantageous to overestimate image-level PE predictions because the metric imposed a prior probability of 1 that the examination was positive for PE. To accommodate this, the sequence head used for refining the image-level PE predictions was trained only on images from examinations positive for PE. The CV-AUC for image-level PE classification in positive examinations only was 0.954 (range, 0.952–0.958). For examination-level labels, CV-AUC ranged from 0.693 for chronic PE to 0.909 for any PE to 0.963 for central PE (Table 2).

Time-Distributed CNN: Adding Model Diversity through Ensembling

Ensembles are a simple and effective method of increasing model performance. Similarly performing models that are less correlated have been shown to yield larger performance increases when ensembled (19). To that effect, an additional model architecture, also built on sequence modeling, was trained to predict the examination-level PE labels. As opposed to the two-staged process outlined above, this model was trained end-to-end, allowing the feature extractor to adapt its feature representations as the sequence model is being trained simultaneously. One caveat of this approach is increased memory requirements; thus, the model input comprised only images in the top 30% of image-level PE predictions using the output from the image-level sequence model above. The model input was first resized to 32 images by 416×416 pixels and cropped in the axial plane to 364×364 pixels.

Ground truth PE image labels were not used to determine which sections to pass as input because during inference only PE model predictions would be available to filter the sections. Thus, out-of-fold PE predictions were used to filter sections to better mimic the input that would ultimately be used during inference. Overall, performance of the end-to-end model was slightly worse across almost all labels than the two-stage sequence model in item 2 above, with CV-AUC ranging from 0.679 for chronic PE to 0.886 for any PE to 0.960 for central PE (Table 2).

RV/LV Ratio Prediction

A separate model was created to predict RV/LV ratio. Predicting PE labels lent itself more naturally to sequence modeling because

Table 2: Fivefold Cross-validation AUC Averages for Examination-Level PE Labels for Model 1 + 2 and Model 3

Description	Model 1 + 2	Model 3
Negative examination for PE	0.909 (0.901–0.916)	0.886 (0.860–0.894)
Indeterminate	0.808 (0.714–0.853)	0.871 (0.813–0.919)
Chronic PE	0.693 (0.654–0.759)	0.679 (0.613–0.715)
Acute and chronic PE	0.855 (0.822–0.891)	0.838 (0.812–0.866)
Central PE	0.963 (0.953–0.971)	0.960 (0.952–0.969)
Left-sided PE	0.908 (0.902–0.916)	0.900 (0.882–0.922)
Right-sided PE	0.928 (0.921–0.939)	0.909 (0.891–0.921)

Note.—Ranges across the five folds are in parentheses. Model 1 and 2 includes image-level feature extraction with sequence modeling and model 3 includes time-distributed convolutional neural network. PE = pulmonary embolism.

image-level annotations were available. RV/LV ratio, on the other hand, was only available at the examination level; in addition, predicting this label lent itself more naturally to 3D models because often the maximum sizes of each ventricle are present on different sections.

For this task, 3D channel-separated networks, originally designed for video modeling, were leveraged, allowing for the use of 3D CNNs pretrained on more than 65 million weakly labeled videos from Instagram (20,21). As mentioned above, pretrained weights provide better initialization for network training, even if the transfer domain is very different from the original task, such as in this case.

Three-dimensional CNNs require a lot of memory to train, often requiring substantial downscaling of native image resolution. Given that a large portion of each CTPA did not contain the heart, images that contained a portion of the heart were manually labeled to train a model that could predict the presence of a “heart image.” Labeling 1000 examinations required approximately 4 hours. A lightweight pruned EfficientNet-B1 model was trained for this simple task, achieving an AUC of 0.999 on single-fold validation. Relevant images containing the heart in each examination could thus be identified and fed as input to the 3D CNN; on average, one-third of images in an examination contained portions of the heart. Rather than resizing an entire CTPA from several hundred sections to 32 images, the heart image identification model substantially reduced the number of input sections to be resized, resulting in less downscaling. In the axial plane, images were resized to 256×256 pixels and cropped to 224×224 pixels, for a model input volume of 32 images by 224×224 pixels.

A peculiarity in labeling RV/LV ratio added to the complexity of the problem. First, rather than a single binary label, there were two binary labels: RV/LV ratio greater than or equal to 1 and RV/LV ratio less than 1. Second, if an examination was negative or indeterminate for PE, both of these labels were 0. To accommodate this during training, all studies negative for PE were reassigned an RV/LV ratio less than 1. The CV-AUC based on this labeling scheme was 0.851.

However, because the presence of PE was clearly related to the final labels, an additional step was incorporated into the final inference process. The above models were used to obtain

examination-level PE predictions (seven classes). Three-dimensional CNNs trained on RV/LV ratio were used to extract features from the input volume, and a final linear model that took both examination-level PE predictions and the extracted 3D CNN features was then trained on the original RV/LV ratio binary labels to mimic the original labeling process. Using extracted features from the 3D CNN improved performance compared with using the final probability prediction alone, reducing the public test set loss from 0.156 to 0.150.

Training details, such as number of epochs, batch size, and choice of optimizer, are not discussed here for the sake of brevity; all specifics can be found at the GitHub repository linked above. In general, batch sizes in the range of 8 to 128 are tested. For optimization, Adam variants such as AdamW or RAdam are preferred (22,23). In this competition, RAdam was more robust than AdamW to the choice of initial learning rate when training transformers. When deciding learning rates, rates in between 10^{-5} and 10^{-3} are tested; optimal learning rates depend on architecture and batch size. Overall, for most applied ML tasks, practitioners are encouraged to focus less on hyperparameter tuning and more on the modeling approach, which typically results in larger gains.

Enforcing Label Consistency Requirements

As mentioned, the competition rules specified label consistency requirements to promote sensible model predictions. While it was possible to modify the loss function to enforce these requirements, it was simpler to apply a heuristic after completion of model inference which modified the final predictions. The heuristic looped through each examination, and if any predictions violated the label consistency requirements, they were modified such that the requirements were no longer violated (eg, if an examination was predicted to be negative for PE ($P < .5$) but the prediction for right-sided PE was greater than 0.5, this prediction was modified to be less than 0.5).

Conclusion

This article describes the second-place approach to the RSNA-STR Pulmonary Embolism Detection challenge. Many of the considerations taken into account are applicable across a wide range of supervised ML problems in medical imaging and beyond.

Although the other top-10 solutions leveraged the same hybrid two-dimensional sequence modeling approach for handling cross-sectional imaging, this approach was unique in its development of a heart image identification model and use of 3D CNNs for RV/LV ratio prediction. In reviewing other top approaches, experimenting with image sizes above the native resolution, as the first-place solution did, would likely have been beneficial. In addition, most top solutions predicted RV/LV ratio jointly with the other PE labels via sequence modeling, which could have improved performance when ensemble with the 3D CNN approach.

Top solutions of ML competitions are often criticized as being too convoluted or computationally intensive to be implemented in the real world. This criticism is not without truth; however, the purpose of ML competitions is not necessarily to provide an out-of-the-box real-world solution. Instead, it allows us to quantify the state of the art and provides an objective platform by which to compare other models. In addition, solutions inspire novel approaches to problems that can be incorporated into real-world solutions.

Overall, the approach to a ML competition is very similar to any kind of model development. One could argue that it is more rigorous and transparent, given the openness of the competition and data controlled by a third-party host, which stands in contrast to much of the work in radiology artificial intelligence today in which data privacy issues can limit one's ability to truly validate the reported results of an article. Competitions also result in the release of rich publicly available datasets that promote further investigation and novel research (24).

Translation of winning competition models to the clinical domain has yet to be fully explored. Future work leveraging the advances made during competitions to ultimately improve patient care and provider satisfaction will be the next step in demonstrating their utility.

Author contributions: Guarantor of integrity of entire study, I.P.; study concepts/study design or data acquisition or data analysis/interpretation, I.P.; manuscript drafting or manuscript revision for important intellectual content, I.P.; approval of final version of submitted manuscript, I.P.; agrees to ensure any questions related to the work are appropriately resolved, I.P.; literature research, I.P.; clinical studies, I.P.; experimental studies, I.P.; statistical analysis, I.P.; and manuscript editing, I.P.

Disclosures of Conflicts of Interest: I.P. Activities related to the present article: disclosed no relevant relationships. Activities not related to the present article: consultant for MD.ai and Centaur Diagnostics. Other relationships: disclosed no relevant relationships

References

- Colak E, Kitamura FC, Hobbs SB, et al. The RSNA Pulmonary Embolism CT Dataset. *Radiol Artif Intell* 2021;3(2):e200254.
- Label Consistency Requirement Details. Kaggle website. <https://www.kaggle.com/c/rsna-str-pulmonary-embolism-detection/discussion/183473>. Accessed January 9, 2021.
- Flanders AE, Prevedello LM, Shih G, et al. Construction of a machine learning dataset through collaboration: The RSNA 2019 Brain CT Hemorrhage Challenge. *Radiol Artif Intell* 2020;2(3):e190211.
- Lee H, Kim M, Do S. Practical Window Setting Optimization for Medical Image Deep Learning. arXiv. [preprint] <https://arxiv.org/abs/1812.00572>. Posted December 3, 2018. Accessed January 2021.
- Wittram C, Maher MM, Yoo AJ, Kalra MK, Shepard JA, McCloud TC. CT angiography of pulmonary embolism: diagnostic criteria and causes of misdiagnosis. *RadioGraphics* 2004;24(5):1219–1238.
- Cubuk ED, Zoph B, Shlens J, Le QV. RandAugment: Practical automated data augmentation with a reduced search space. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2020; 3008–3017.
- Shih G, Wu CC, Halabi SS, et al. Augmenting the National Institutes of Health chest radiograph dataset with expert annotations of possible pneumonia. *Radiol Artif Intell* 2019;1(1):e180041.
- Micikevicius P, Narang S, Alben J, et al. Mixed Precision Training. 6th International Conference on Learning Representations, 2018.
- Tan M, Le QV. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. Proceedings of the 36th International Conference on Machine Learning 2019; 97:6105–6114. <http://proceedings.mlr.press/v97/tan19a.html>. Accessed January 2021.
- Zhang H, Wu C, Zhang Z, et al. ResNeSt: Split-Attention Networks. arXiv website. <https://arxiv.org/abs/2004.08955>. Posted April 19, 2020. Updated December 30, 2020. Accessed January 9, 2021.
- PyTorch Image Models. GitHub website. <https://github.com/rwightman/pytorch-image-models>. Accessed January 9, 2021.
- Radenovic F, Tolias G, Chum O. Fine-Tuning CNN Image Retrieval with No Human Annotation. *IEEE Trans Pattern Anal Mach Intell* 2019;41(7):1655–1668.
- Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Comput* 1997;9(8):1735–1780.
- Cho K, van Merriënboer B, Gulcehre C, et al. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, 2014; 1724–1734.
- Vaswani A, Shazeer N, Parmar N, et al. Attention Is All You Need. Proceedings of the 31st International Conference on Neural Information processing Systems, 2017; 6000–6010. <https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf>. Accessed January 2021.
- Dosovitskiy A, Beyer L, Kolesnikov A, et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. arXiv [preprint]. <https://arxiv.org/abs/2010.11929>. Posted October 22, 2020. Accessed January 2021.
- Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, Zagoruyko S. End-to-End Object Detection with Transformers. arXiv [preprint]. <https://arxiv.org/abs/2005.12872>. Posted May 26, 2020. Updated May 28, 2020. Accessed January 2021.
- Wolf T, Debut L, Sanh V, et al. Transformers: State-of-the-Art Natural Language Processing. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, 2020; 38–45. <https://aclanthology.org/2020.emnlp-demos.6>. Accessed January 2021.
- Pan I, Thodberg HH, Halabi SS, Kalpathy-Cramer J, Larson DB. Improving automated pediatric bone age estimation using ensembles of models from the 2017 RSNA machine learning challenge. *Radiol Artif Intell* 2019;1(6):e190053.
- Tran D, Wang H, Torresani L, Feiszli M. Video Classification with Channel-Separated Convolutional Networks. 2019 IEEE/CVF International Conference on Computer Vision, 2019; 5551–5560.
- Ghadiyaram D, Feiszli M, Tran D, Yan X, Wang H, Mahajan D. Large-scale weakly-supervised pre-training for video action recognition. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019; 12038–12047.
- Loshchilov I, Hutter F. Decoupled Weight Decay Regularization. arXiv [preprint]. <https://arxiv.org/abs/1711.05101>. Posted November 14, 2017. Updated January 4, 2019. Accessed January 2021.
- Liu L, Jiang H, He P, et al. On the Variance of the Adaptive Learning Rate and Beyond. arXiv [preprint]. <https://arxiv.org/abs/1908.03265>. Posted August 8, 2019. Updated April 17, 2020. Accessed January 2021.
- Prevedello LM, Halabi SS, Shih G, et al. Challenges Related to Artificial Intelligence Research in Medical Imaging and the Importance of Image Analysis Competitions. *Radiol Artif Intell* 2019;1(1):e180031.