Ensembled Transfer Learning for MRI Injury Prediction

Shayne Miel
Stanford University
Stanford, CA, USA
smiel@stanford.edu

Abstract

Improving the accuracy with which automated methods can identify injuries in MRIs of the knee would lead to time and cost savings for doctors and patients. I show that by ensembling a collection of transfer learning models that use pretrained neural networks for feature extraction, as well as a mixture of max pooling and an attention mechanisms to combine features from multiple frames of the MRI scan, I can improve upon the current state of the art on the MRNet data set.

1. Introduction

Magnetic resonance imaging (MRI) is a method of obtaining three dimensional images of the inside of an object by using strong magnets to align the protons in the object and then radio frequency currents to disrupt and measure that alignment. An MRI sequence is a series of two dimensional images that can be stacked to recreate the three dimensional object. There are three sequence types, each corresponding to an orientation of the "camera" that captures the two dimensional slices. The axial sequence captures slices of the object that are horizontal to the ground; the coronal sequence captures vertical slices as viewed from the front of the object; and the sagittal sequence captures vertical slices as viewed from the side of the object.

MRI can be a useful tool when diagnosing knee injuries[19, 8, 23], however, analyzing the images is a time-consuming process and even with trained professionals, it is easy for a clinician to misdiagnose an injury based on an MRI reading[14]. Improving the automated identification of abnormalities in knee MRIs could help prioritize which MRIs to examine first, as well as provide better early results for patients whose scans appear normal. Model predictions could also provide a "second opinion" which would reduce the possibility of missed abnormalities. This could represent a large cost savings for hospitals and an increased level of care for patients.

The problem addressed in this paper is as follows: given a set of three MRI sequences (axial, coronal, and sagittal) of a patient's knee, can we predict the presence of injuries that will require surgery? In particular, we wish to predict whether the knee is healthy, has an ACL tear, has a meniscal tear, or has any other abnormality. Since these injuries can co-occur, we wish to predict three independent binary values: abnormal, acl tear, and meniscus tear. Past research has proven that MRI scans are accurate and sensitive tools for detecting these kinds of injuries in a non-invasive manner[4, 7, 27].

2. Related work

There are two main challenges when trying to use automated methods on MRI sequences. The first is that each MRI image has high information content because it is a 3-dimensional image of the patient. Determining how to extract meaning from a variable length series of images is not trivial. By contrast, the second challenge is that MRIs are difficult and expensive to collect, which means that MRI data sets are too small for most deep learning approaches. This means that deep learning models must be constrained to a limited number of updates to prevent overfitting.

Most recent deep learning MRI work deals with the 3-dimensional nature of MRI images by using 3D-CNNs, which take the idea of $(channel, width, height)$ convolutions that are swept across a 2-dimensional image and extends them to $(time, channel, width, height)$ convolutions that are swept across the 3-dimensional image [11, 26, 13, 29]. These models are limited, however, by the number of parameters required for 3D-CNNs and the need to train them from scratch.

Another common method is the so-called 2.5D approach, where models are trained on sliding windows of the input sequence [22, 2, 9]. This works well for sequence-to-sequence problems, but doesn't help when you need to reduce the MRI to a single prediction.

RNNs provide yet another avenue for processing a series of 2-D images, as done in [21, 10]. However, the need to train from scratch and the number of parameters in these kinds of models still presents a very real challenge for small

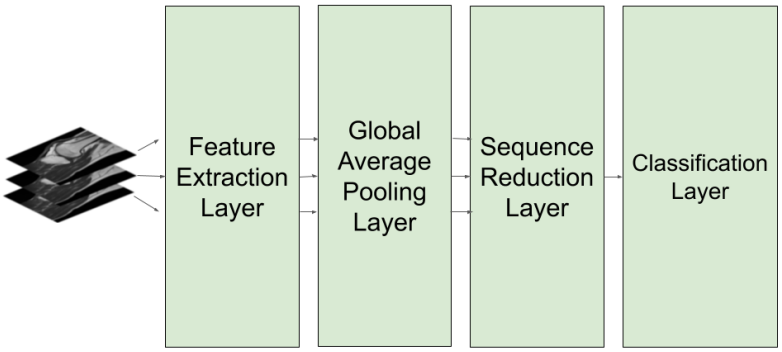

Figure 1. An abstract view of the sequence-specific networks.

data sets.

The closest model to the ones in this paper is Bien et al.'s MRNet[3], which is the current state-of-the-art MRNet for diagnosing injuries in MRI scans of the knee. Their method uses conventional convolutional neural networks, which allows them to jump start the training process by pretraining an AlexNet[15] network on a large data set like ImageNet[5]. The sequence of images are each fed through the pretrained network to generate an $(s \times c \times w \times h)$ tensor, where s is the sequence length, c is the channel depth, w is the width and h is the height. The width and height dimensions are reduced via global average pooling[16], so that they are left with an $(s \times c)$ tensor. Those features are then reduced via max pooling over the sequence and fed through a linear classifier and sigmoid activation to predict a probability for the sequence. Three sequence-level probabilities (axial, coronal, and sagittal) are then sent to a logistic regression classifier to obtain a final diagnosis probability.

3. Methods

3.1. Sequence Network

Figure 1 shows an abstract process for turning a single MRI sequence into an injury prediction. On the left side of the diagram, a series of n images, each a $\mathbb{R}^{3\times224\times224}$ matrix, are fed into the network as a single batch. The feature extractor converts each of these images into a $\mathbb{R}^{c\times w\times h}$ matrix, where c is the number of channels and w and h are the width and height, respectively. These features are then flattened into a series of $\mathbb{R}^c$ vectors in the global average pooling layer. Finally, the full batch of n vectors are flattened into a single $\mathbb{R}^c$ vector in the sequence reduction layer. This vector is then passed through a linear layer and a sigmoid activation in the classification layer to arrive at a probability for whether the entire MRI sequence indicates the injury in question.

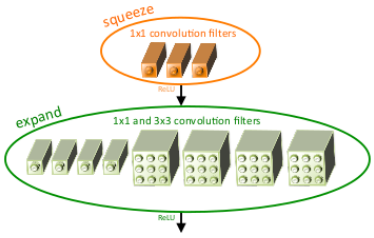

Figure 2. Illustration of a fire layer. Taken directly from "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size"[12]

Concretely, if we describe the feature extraction layer as a function, $f : \mathbb{R}^{n\times3\times244\times244} \to \mathbb{R}^{n\times c\times w\times h}$, and the sequence reduction layer as a function $g : \mathbb{R}^{n\times c} \to \mathbb{R}^c$, then the entire network can be written as:

$$gap_{jk}(x) = \frac{\sum_{i=1}^{n} x_{j,k}^{(i)}}{n}$$

$$p = \sigma(g(gap(f(x)))W_c + b_c) \tag{1}$$

where p is the predicted probability, $W_c \in \mathbb{R}^{c\times1}$ and $b_c \in \mathbb{R}^c$ are the weights and biases of the classification layer and σ is the sigmoid function: $\frac{1}{1+e^{-x}}$

The current state-of-the-art system, MRNet - which was discussed in detail in Section 2, can be described as using parts of a pretrained AlexNet for the feature extractor layer, and an element-wise maximum to flatten the sequence in the sequence reduction layer.

3.2. SqueezeNet

SqueezeNet is a deep convolutional neural network that achieves high performance on the ImageNet challenge. It is composed of a series of "fire" layers, each of which consists

of a layer of 1×1 convolutional filters (the squeeze layer), followed by a ReLU and then a mix of 1×1 and 3×3 convolutional layers (the expand layer) whose outputs are concatenated before going through a final ReLU activation. An illustration of a fire layer is shown in Figure 2. These layers are interspersed with max pooling layers before the 1st, 4th, and 8th fire layers, and the entire network begins with a traditional 7×7 convolutional layer.

Fine-tuning a SqueezeNet model that has been pretrained on the ImageNet data set is a method that has been used successfully to classify vehicles[1], crop disease[6], and cataracts[20]. By removing the final dropout, convolution, ReLU and adaptive average pooling layers from the pretrained network, we can use it as a drop-in replacement for the feature extraction layer in Figure 1.

3.3. Attention

Attention is a mechanism that has traditionally been applied as a summarization method of the hidden states in a recurrent neural network. Given some query **Q**, a set of keys **K**, and a set of values **V**, the attention score can be calculated as

$$s = (\mathbf{Q}\mathbf{K}^T)\mathbf{V}$$

$$\mathbf{A}_i = \frac{e^{s_i}}{\sum_{j=1}^{k} e^{s_j}}$$

as described by Tan, et al.[25], whose work was derived from Luong et. al's location-based attention[17].

In the case of an RNN, the query is often the final hidden state, the keys are either the input embeddings or the hidden states at each time step, and the values are the hidden states at each time step. A final representation of the entire sequence can then be generated by taking the weighted sum of the values times the attention vector, $result = \mathbf{A}\mathbf{V}^T$.

For the experiments in this paper, we eschew the use of an RNN and simply use the attention-weighted sum across frames of the MRI sequence to generate the final representation of the sequence. In this case, the query is a learned parameter of the network, while the keys and values are both the output of the global average pooling layer for each frame in the sequence. This allows us to use the attention-weighted sum as the sequence-reduction layer:

$$g(x) = \mathbf{A}x^T$$

Because of the imbalanced class sizes, we train all of the sequence-specific models with a weighted binary cross-entropy loss:

$$L = -\frac{1}{N}\sum_{i=0}^{N} w_i\big(y_i log(p_i) + (1 - y_i) log(1 - p_i)\big)$$

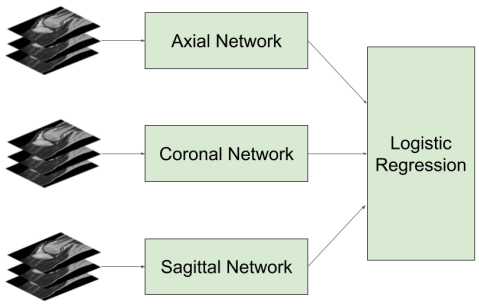

Figure 3. Ensembling predictions from the sequence-specific networks to generate an injury prediction.

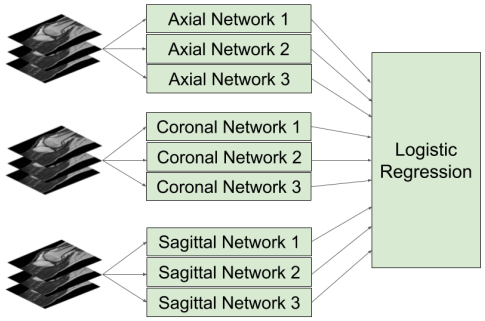

Figure 4. Ensembling predictions from multiple sequence-specific network types to generate an injury prediction.

where w_i is the fraction of the instances in the data set that have label y_i, and p_i is the output of the network described by Equation 1 for sequence i.

3.4. Sequence Ensembling

Because we have access to multiple MRI sequences per patient, separate networks can be trained for each of the axial, coronal, and sagittal sequences. The three probabilities are then fed into a logistic regression classifier to arrive at a final injury prediction, as shown in Figure 3.

3.5. Model and Sequence Ensembling

In addition to this per-model sequence ensembling, we can also use the sequence predictions from *multiple model types* as features for a logistic regression model, as shown in Figure 4, as a meta-ensemble.

4. Dataset

The MRI data provided in the MRNet challenge contains scans from 3 MRI types (axial, coronal, and sagittal) with 3 labels per MRI (abnormality, ACL tear, and meniscal tear) for 1,250 examinations.

The released data has already been split into a training and validation set, and a test set has been withheld for

Diagnosis	Label	Training	Tuning	Validation
Abnormal	Positive	835	78	95
	Negative	175	42	25
	Total	1010	120	120
ACL	Positive	167	41	54
	Negative	843	79	66
	Total	1010	120	120
Meniscus	Positive	353	44	52
	Negative	657	76	68
	Total	1010	120	120

Table 1. MRNet data splits and label counts.

leaderboard purposes. In order to evaluate my methods for this paper, I have further divided the training set into training and tuning sets, and am using the validation set for all reported metrics. Counts of cases and labels for each set can be seen in Table 1.

4.1. Preprocessing

The data is provided as a collection of numpy[18] 3-dimensional matrices, one per sequence per case. They have already been extracted from the Digital Imaging and Communications in Medicine (DICOM) files and scaled to 256×256 images.

During training, we crop the images to 224×224 to match the expected input size for a pretrained ImageNet model. We then standardize each input sequence by subtracting the minimum pixel value and dividing by the range observed in the sequence, then rescale the image to a $0-256$ range of pixel values. Finally, we normalize the sequence by subtracting the data set mean (58.09) and dividing by the data set standard deviation (49.73).

4.2. Augmentation

In order to prevent overfitting the small data set, we perform data augmentation during training. Every image is randomly flipped horizontally, shifted horizontally by -25 to 25 pixels, and rotated by -25 to 25 degrees each time it is seen during training. Note that the same transformation is applied to every image in a given sequence.

5. Results

Referring to the abstract network image shown in Figure 1, Table 2 shows the combinations of feature extraction and sequence reduction techniques used in the experiments in this paper. In each case, a pretrained network that was tuned on ImageNet was used to extract features. The pretrained network was fine-tuned on the MRNet data along with the other layers of the model.

In addition to these models, I include results from ensembling the sequence-specific predictions of all four models, as described in Figure 4.

Table 3 shows the hyperparameters used for all experiments. Most of these hyperparameters were drawn from the original MRNet paper, although Initial Learning Rate and Max Epochs were determined with brief experiments on the axial sequence and abnormal label. Due to the processing time required to run these models, extensive hyperparameter optimization was not feasible, although this would be a potential are for future work.

The primary metric used to compare models is Area Under the ROC Curve (AUC). In particular, averaging the AUC of the three labels determines a final metric for each model. In addition, I report the specificity, sensitivity, and accuracy of the four models on each label, using a 0.5 threshold to turn probabilities into label predictions.

Table 4 shows the AUC for each model on each injury prediction, as well as averaged across all injuries. Each of the individual model types perform well on some of the injury categories, but none of them excel at all injuries. For detecting any kind of abnormality, the original MRNet implementation gets the highest AUC (0.940). MRNet-Squeeze is the best model for detecting ACL injuries (0.974), while MRNet-SqueezeAttend achieves the highest AUC on the Meniscal tear category (0.885).

By using all four model types together, however, I can capture that varying performance on the three injuries to create an ensembled model that outperforms any of the individual models. The ensembled model achieves nearly the highest AUC on all individual injury types, and significantly outperforms the others with an average AUC of 0.931.

MRNet (reported) is the AUC reported by Bien, et al.[3], and is not directly comparable to the values shown here because the released data is a subset of the data used in their experiments.

Table 5 shows the specificity, sensitivity, and accuracy of each model, as well as the human performance of expert radiologists, as reported by Bien et al.[3]. Once again, the experiments in this paper are not directly comparable to the reported human values because of differing data sets, especially on these metrics since class imbalance can have an outsized effect on precision and recall. On these metrics, the final ensemble achieves state of the art on some injury types and not others.

It is interesting to consider the differences between the four models, especially at the level of the sequence-specific predictions (i.e. the values that get fed into the logistic regression classifier). If the sequence-specific predictions for each diagnosis are highly correlated, then we would not expect the multi-model ensemble to outperform the individual models. Figure 5 shows the correlation between the sequence-specific predictions on the validation set for the four models. There is a strong correlation between the sequence-specific predictions on some diagno-

Name	Feature Extraction Layer	Sequence Reduction Layer
MRNet	AlexNet	Max pooling
MRNet-Squeeze	SqueezeNet	Max pooling
MRNet-Attend	AlexNet	Attention
MRNet-SqueezeAttend	SqueezeNet	Attention

Table 2. Model combinations used in experiments.

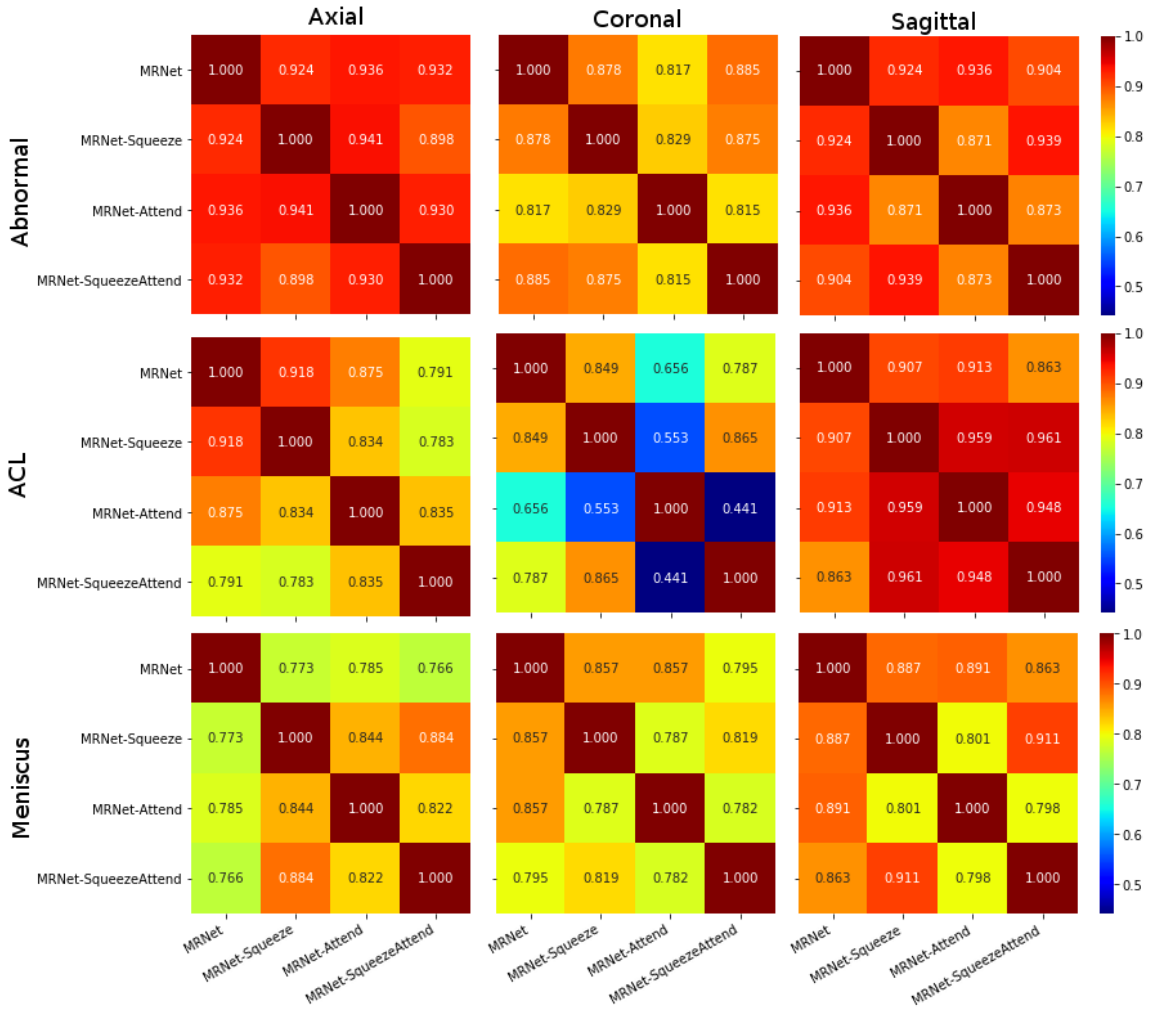

Figure 5. Correlation between sequence-specific predictions on the validation set.

sis/sequence combinations (for instance, Abnormal/Axial and ACL/Sagittal), but on other combinations the correlation is quite low (ACL/Coronal, Meniscus/Axial). These areas of low correlation are likely useful signals for the multi-model ensemble.

Another way to look at the differences between models is to examine, for a particular diagnosis/sequence combination, what areas of the MRI each model is focusing on. If we consider a specific instance in the validation set, Case 1218, and look at a specific diagnosis/sequence pair, ACL/Axial, we can ask which image from the sequence each model is most interested in. For the models that use attention (MRNet-Attend and MRNet-SqueezeAttend) that simply means asking for the argmax of the calculated attention vector. For MRNet and MRNet-Squeeze, the sequence reduction layer takes the maximum value over the sequence for each location in the features vector. For those models, we can say that the frame with the most maximum values

Optimization Method	Adam
Weight Decay	0.01
Learning Rate Schedule	Reduce on plateau
Initial Learning Rate	0.00001
Max Patience	5
Factor	0.3
Max Epochs	40
Logistic Regression Penalty	L2
Logistic Regression Lambda	1.0

Table 3. Hyperparameters for all experiments.

is the one that the model is "most interested" in. Table 6 shows the most interesting frame of the axial sequence for each model when asked to predict whether Case 1218 has an ACL tear.

We can go further though, and examine the class activation map (CAM) for each of those frames for each model. A class activation map is a way of visualizing where in the image a model is focusing its (implicit) attention[28]. To calculate the CAM for the frames listed in Table 6, we use W_c from Equation 1 to compute a weighted average of the c $\mathbb{R}^{w \times h}$ feature maps generated by running the single frame through the feature extraction layer of the sequence-specific model. That is, let F be the $\mathbb{R}^{c \times w \times h}$ tensor returned by running a frame through the feature extraction layer. Then we calculate an image $I \in \mathbb{R}^{w \times h}$:

$$I_{i,j} = \frac{1}{c} \sum_{a=1}^{c} W_{c_a} F_{i,j}$$

By scaling this image up to the original MRI image size (256×256) and using the values as a colormap to overlay on the image, we can create a visualization of where the model's attention is focused for each frame. Figure 6 shows the CAM for the four models (across the columns) for the 11th, 7th, 20th, and 6th frame, i.e. each model's "most interesting" frame (across the rows).

6. Conclusion/Future Work

The experiments in this paper use transfer learning to predict injury diagnoses from MRI sequences of the knee. I compare pretrained AlexNet and SqueezeNet models as feature extractors and max pooling versus attention mechanisms to reduce the information across the sequence. Finally, I use a logistic regression, trained on the sequence-specific predictions, to predict the probability of the diagnosis for the patient.

I show that all of the models have different strengths and weaknesses, some achieving higher AUC scores than others on each diagnosis. Furthermore, I show that by ensembling not just different sequence predictions, but multiple sequence predictions from multiple sequence models, I can outperform the current state of the art on the MRNet data set.

There is still a lot left to explore here though. Future work should include examining how to get greater diversity in the sequence-specific models. Some possibilities for decreasing the correlation between predictions are:

- Using other pretrained feature extractors in the Feature Extraction layer.
- Swapping the order of the Global Average Pooling layer and the Sequence Reduction layer.
- Using more advanced methods like RNNs in the Sequence Reduction layer.

I also believe that there is a possibility of using inter-sequence attention to compute a final representation of all three sequences, but exploring that was beyond the scope of this project. Finally, an end-to-end ensemble model, instead of using logistic regression on the sequence-specific predictions, may lead to more robust models.

6.1. Failed Experiments

I briefly tried some of the next steps mentioned above, but was not able to achieve AUC scores as high as any of the individual models, much less the final multi-model ensemble. With more time, any one of these paths could be useful.

Specifically, I tried:

- Using a pretrained ResNet in place of AlexNet or SqueezeNet. Despite many attempts at freezing or removing various layers and adding dropout, I was not able to prevent models with feature extractor from overfitting on the sequence-specific predictions.
- Using pretrained DenseNet or VGG in place of AlexNet or SqueezeNet. DenseNet and VGG both require significantly more memory than the other networks, which prevented me from using them since the model architecture requires passing all of the frames in an MRI sequence through at once.
- Using LSTM and BiLSTM layers in place of max pooling or attention in the Sequence Reduction layer. Here I struggled with underfitting, even with more LSTM layers.
- Training end-to-end networks instead of ensembling the sequence-specific predictions. I believe that these networks were overfitting. The average AUC scores ranged from 0.860 (MRNet-SqueezeAttend) to 0.889 (MRNet).

Model	Average	Abnormal	ACL	Meniscus
MRNet (reported)	0.916	0.937	0.965	0.847
MRNet	0.913	**0.940**	0.960	0.839
MRNet-Squeeze	0.910	0.925	0.974	0.829
MRNet-Attend	0.891	0.925	0.910	0.838
MRNet-SqueezeAttend	0.915	0.936	0.925	**0.885**
Ensemble	**0.931**	0.939	**0.976**	0.876

Table 4. AUC on the validation set

Prediction	Specificity	Sensitivity	Accuracy
Abnormality			
Radiologist	0.844	0.905	0.894
MRNet (reported)	0.714	0.879	0.850
MRNet	0.440	0.968	0.858
MRNet-Squeeze	0.560	0.968	0.883
MRNet-Attend	0.480	0.979	0.875
MRNet-SqueezeAttend	0.440	0.968	0.858
Ensemble	0.480	0.958	0.858
ACL tear			
Radiologist	0.933	0.906	0.920
MRNet (reported)	0.968	0.759	0.867
MRNet	0.894	0.907	0.900
MRNet-Squeeze	0.909	0.963	0.933
MRNet-Attend	0.803	0.778	0.792
MRNet-SqueezeAttend	0.864	0.852	0.858
Ensemble	0.909	0.981	0.942
Meniscus tear			
Radiologist	0.882	0.820	0.849
MRNet (reported)	0.741	0.710	0.725
MRNet	0.721	0.788	0.750
MRNet-Squeeze	0.735	0.731	0.733
MRNet-Attend	0.691	0.846	0.758
MRNet-SqueezeAttend	0.794	0.750	0.775
Ensemble	0.735	0.865	0.792

Table 5. Metrics on the validation set

Model	Frame
MRNet	11
MRNet-Squeeze	7
MRNet-Attend	20
MRNet-SqueezeAttend	6

Table 6. The most focused on frame of the axial sequence for each model when predicting whether Case 1218 has an ACL tear.

A. Contributions & Acknowledgements

Compute power for all experiments and time to work on the project generously donated by Turnitin. I would also like to thank Stanford cs231n TA, David Morales, for his advice and suggestions as I completed this work.

B. Project Code

All code, notebooks, images, and LaTeX for this paper can be found in this GitHub repo.

C. Starter Code

Bien et al. supply code for training and evaluating MRNet on a set of external validation data from the 2017 Štajduhar et al. study [24]. The model.py file in this package provided the architecture of the baseline MRNet model. I also used the train.py, evaluate.py, and loader.py files as starting places for my work, though they had to be heavily modified to work on the MRNet challenge data.

The GitHub repo linked above contains a copy of the original code in the 'external_validation_scripts' directory.

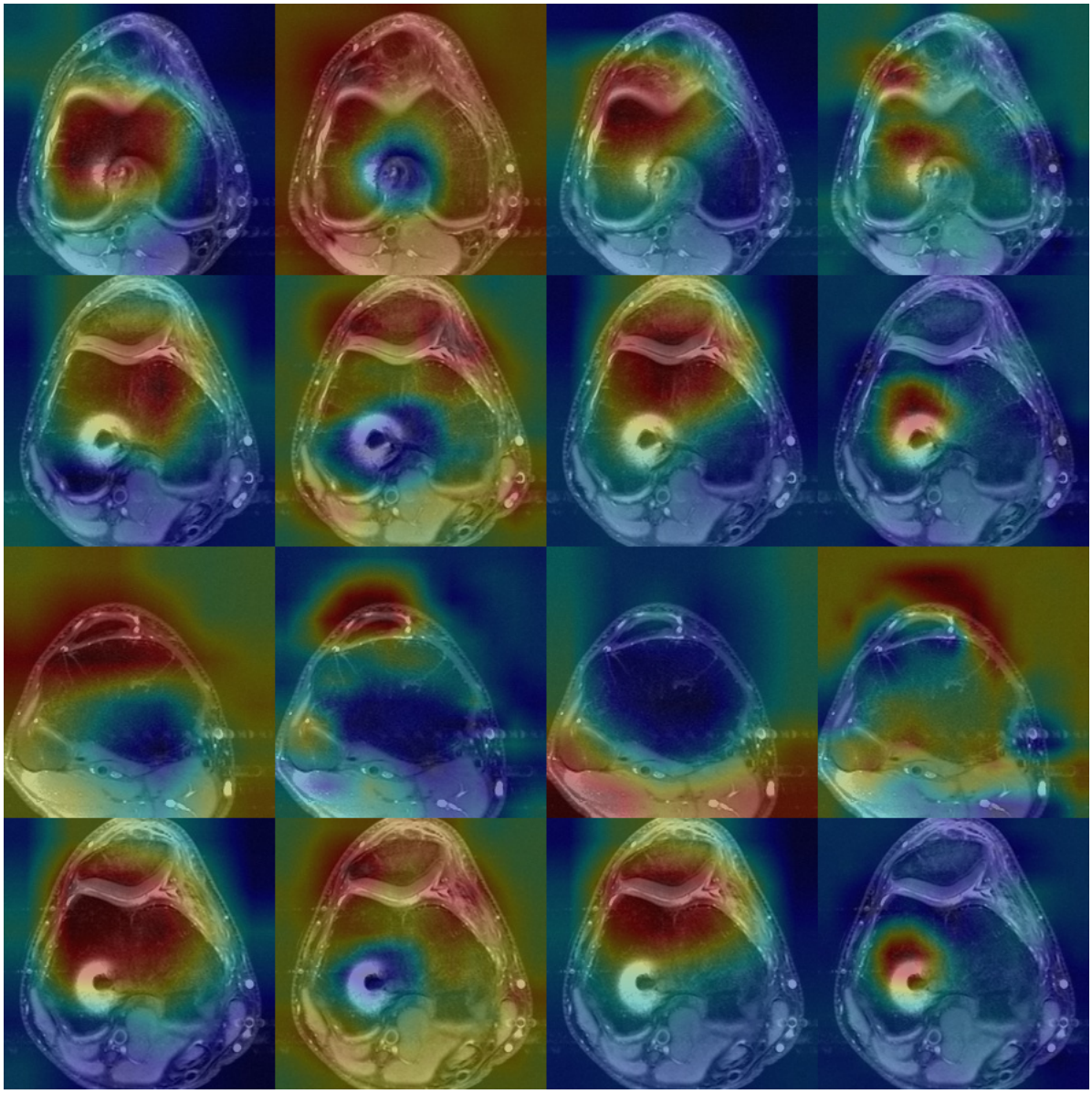

Figure 6. Class activation maps from the four networks for the axial sequence of Case 1218, when asked to predict whether there is an ACL tear. From left to right, the columns are class activation maps for MRNet, MRNet-Squeeze, MRNet-Attend, and MRNet-SqueezeAttend. Each row represents the frame from the axial sequence that each network found most interesting.

References

[1] A. S. Agoes, Z. Hu, and N. Matsunaga. Fine tuning based squeezenet for vehicle classification. In *Proceedings of the International Conference on Advances in Image Processing*, pages 14–18. ACM, 2017.

[2] R. Alkadi, A. El-Baz, F. Taher, and N. Werghi. A 2.5 d deep learning-based approach for prostate cancer detection on t2-weighted magnetic resonance imaging. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018.

[3] N. Bien, P. Rajpurkar, R. L. Ball, J. Irvin, A. Park, E. Jones, M. Bereket, B. N. Patel, K. W. Yeom, K. Shpanskaya, et al. Deep-learning-assisted diagnosis for knee magnetic resonance imaging: Development and retrospective validation of mrnet. *PLoS medicine*, 15(11):e1002699, 2018.

[4] B. F. Boeve, R. Davidson, and J. E. Staab. Magnetic resonance imaging in the evaluation of knee injuries. *Southern medical journal*, 84(9):1123–1127, 1991.

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and

L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[6] H. Durmuş, E. O. Güneş, and M. Kırcı. Disease detection on the leaves of the tomato plants by using deep learning. In *2017 6th International Conference on Agro-Geoinformatics*, pages 1–5. IEEE, 2017.

[7] L. Felli, G. Garlaschi, A. Muda, A. Tagliafico, M. Formica, A. Zanirato, and M. Alessio-Mazzola. Comparison of clinical, mri and arthroscopic assessments of chronic acl injuries, meniscal tears and cartilage defects. *Musculoskeletal surgery*, 100(3):231–238, 2016.

[8] S. Figueiredo, L. S. Castelo, A. D. Pereira, L. Machado, J. A. Silva, and A. Sa. Use of mri by radiologists and orthopaedic surgeons to detect intra-articular injuries of the knee. *Revista brasileira de ortopedia*, 53(1):28–32, 2018.

[9] E. Grøvik, D. Yi, M. Iv, E. Tong, D. L. Rubin, and G. Zaharchuk. Deep learning enables automatic detection and segmentation of brain metastases on multi-sequence mri. *arXiv preprint arXiv:1903.07988*, 2019.

[10] L. Han and M. R. Kamdar. Mri to mgmt: predicting methylation status in glioblastoma patients using convolutional recurrent neural networks. In *Pac Symp Biocomput*, volume 23, pages 331–42. World Scientific, 2018.

[11] E. Hosseini-Asl, G. Gimel'farb, and A. El-Baz. Alzheimer's disease diagnostics by a deeply supervised adaptable 3d convolutional network. *arXiv preprint arXiv:1607.00556*, 2016.

[12] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and¡ 0.5 mb model size. *arXiv preprint arXiv:1602.07360*, 2016.

[13] A. Khvostikov, K. Aderghal, J. Benois-Pineau, A. Krylov, and G. Catheline. 3d cnn-based classification using smri and md-dti images for alzheimer disease studies. *arXiv preprint arXiv:1801.05968*, 2018.

[14] G. Kolata. Sports medicine said to overuse m.r.i.'s, October 2011. [Online; posted 28-October-2011].

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[16] M. Lin, Q. Chen, and S. Yan. Network in network. *arXiv preprint arXiv:1312.4400*, 2013.

[17] M.-T. Luong, H. Pham, and C. D. Manning. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*, 2015.

[18] T. E. Oliphant. A guide to numpy, 2006–. [Online; accessed 02-June-2019].

[19] N. Orlando Júnior, M. G. d. S. Leão, and N. H. C. d. Oliveira. Diagnosis of knee injuries: comparison of the physical examination and magnetic resonance imaging with the findings from arthroscopy. *Revista brasileira de ortopedia*, 50(6):712–719, 2015.

[20] X. Qian, E. W. Patton, J. Swaney, Q. Xing, and T. Zeng. Machine learning on cataracts classification using squeezenet. In *2018 4th International Conference on Universal Village (UV)*, pages 1–3. IEEE, 2018.

[21] C. Qin, J. Schlemper, J. Caballero, A. N. Price, J. V. Hajnal, and D. Rueckert. Convolutional recurrent neural networks for dynamic mr image reconstruction. *IEEE transactions on medical imaging*, 38(1):280–290, 2019.

[22] H. R. Roth, L. Lu, J. Liu, J. Yao, A. Seff, K. Cherry, L. Kim, and R. M. Summers. Improving computer-aided detection using convolutional neural networks and random view aggregation. *IEEE transactions on medical imaging*, 35(5):1170–1181, 2015.

[23] T. Smith, M. Lewis, F. Song, A. Toms, S. Donell, and C. Hing. The diagnostic accuracy of anterior cruciate ligament rupture using magnetic resonance imaging: a meta-analysis. *European Journal of Orthopaedic Surgery & Traumatology*, 22(4):315–326, 2012.

[24] I. Štajduhar, M. Mamula, D. Miletić, and G. Ünal. Semi-automated detection of anterior cruciate ligament injury from mri. *Computer methods and programs in biomedicine*, 140:151–164, 2017.

[25] Z. Tan, M. Wang, J. Xie, Y. Chen, and X. Shi. Deep semantic role labeling with self-attention. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[26] W. Wei, E. Poirion, B. Bodini, S. Durrleman, O. Colliot, B. Stankoff, and N. Ayache. Flair mr image synthesis by using 3d fully convolutional networks for multiple sclerosis. In *ISMRM-ESMRMB 2018-Joint Annual Meeting*, pages 1–6, 2018.

[27] J. Yaqoob, M. S. Alam, and N. Khalid. Diagnostic accuracy of magnetic resonance imaging in assessment of meniscal and acl tear: Correlation with arthroscopy. *Pakistan journal of medical sciences*, 31(2):263, 2015.

[28] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016.

[29] L. Zou, J. Zheng, C. Miao, M. J. Mckeown, and Z. J. Wang. 3d cnn based automatic diagnosis of atten-

tion deficit hyperactivity disorder using functional and structural mri. *IEEE Access*, 5:23626–23636, 2017.