Detecting knee abnormalities from MRI images

Sarah Egler
Stanford University
segler@stanford.edu

Mara Finkelstein
Stanford University
mfinkels@stanford.edu

Giacomo Lamberti
Stanford University
giacomol@stanford.edu

Abstract

Magnetic Resonance Imaging (MRI) is commonly used to detect knee abnormalities; however, diagnoses can be time-consuming and subject to the interpretation of radiologists. In the present work, we employ deep learning models to detect 1) general knee abnormalities, 2) anterior cruciate ligament (ACL) tears, and 3) meniscal tears from MRI exams. We extend upon MRNet [1], a convolutional neural network (CNN) architecture which trains three separate models for these same tasks, with a single Multi-task model. We also take advantage of the series dimension in MRI exams and use long short-term memory (LSTM) cells as well as 3D CNNs. The multi-task model achieves an average AUC score of 0.88 on the three tasks, while reducing the computational cost compared to MRNet. It outperforms the LSTM and 3D CNN architectures, perhaps due to the smaller number of parameters afforded by a global average pooling layer and lower propensity for overfitting a relatively small dataset. Across all models, the best performance was on the general abnormality detection task.

1. Introduction

The knee is the largest and most complex joint in the human body, and is also one of the most injured [7]. Two common knee injuries occur in the anterior cruciate ligament (ACL) and the meniscus. A sketch of a knee, including the location of the ACL and meniscus, is shown in Figure 1.

Magnetic Resonance Imaging (MRI), which employs a strong magnetic field to produce detailed images of organs, tissues and bones, is commonly used to detect knee abnormalities, including meniscal and ACL tears [12]. During a MRI exam, three sequences of several MRI images, corresponding to three different views (i.e. axial, coronal and sagittal), are generated. A sagittal series, for example, would begin on the side of the body at the outside of the knee, with each slice moving in towards the inside of the knee, essentially creating a 3D image of the knee. Generally, the more relevant information is contained within the innermost slices [7]. The great detail of information deriving from these exams makes the interpretation of the results time-consuming and subject to misdiagnosis. On the other hand, computer-aided diagnoses (CAD) have the potential to reduce the risk of misdiagnosis of knee abnormalities and ensure timely care to all patients. Within the CAD toolbox, deep learning represents a powerful technique, given its ability to learn extremely complex functions and outperform other methods in computer vision applications.

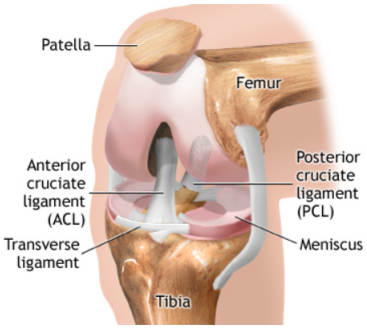


Figure 1: Knee anatomy (Image courtesy of *The Chelsea knee clinic*)

The objective of this project is to develop a deep learning model to detect knee injury using MRI exams. Specifically, given an input MRI exam, where each exam is a sequence of images, we will output three independent binary predictions, one for each the following tasks: 1) general abnormality, 2) ACL tear, and 3) meniscal tear.

For example, an exam may be labeled as 1 for general abnormality, 1 for ACL tear, and 0 for meniscal tear. The objective and dataset for this work come from the MRNet Competition hosted by the Stanford ML Group [1].

2. Related Work

2.1. MRNet

In recent years, traditional computer vision deep learning models have been successfully extended to the task of interpreting MRI images. The MRNet model, developed in [1], is a fully automated deep learning model to detect knees abnormalities. MRNet uses AlexNet [9] to extract

features from each image of a MRI exam, then a global average pooling, max-pooling and a fully connected layer to output a binary prediction. A separate model is trained for each of the three tasks (general abnormality, ACL tear, and meniscal tear) and views (axial, coronal, and sagittal), for a total of nine models. For each task, the predictions from all of the views are combined using logistic regression to produce a single output probability. The model is able to provide specific diagnoses of ACL and meniscal tears with accuracy comparable to that of practicing radiologists. We believe this model can be extended to take advantage of spatiotemporal features inherent to MRI exam sequences, as well as consolidated into a multitask approach rather than training separate models for each task.

2.2. Spatiotemporal Approaches

Architectures such as [14] and [5] have taken advantage of spatial contextual information in MRI volumes by using 3D convolutional neural networks, replacing the traditional three color channels with slices of the volume. [14] found that 3D convolutions outperformed 2D convolutions in predicting Alzheimers Disease from brain MRI exams. Moreover, relevant to knee anatomy and the task at hand, 3D CNNs were used to detect degenerative meniscus and cartilage changes in osteoarthritic and ACL subjects [15]. Together, these similar use cases drive our approach in the application of 3D CNNs to the current problem. Driven by the complexity of 3D CNNs and their propensity to overfit, the authors of [18] cleverly define a so-called top-heavy 3D CNN. They replace the bottom layers of a fully 3D CNN with 2D convolutional layers, and find that this yields better performance in time, while also reducing the number of parameters. This suggests that the temporal component of learning in 3D networks may be effective on higher-level representations, which is of interest given the small size of our dataset (leading to concerns of overfitting) and our limited compute resources.

The third dimension of MRI images can also be processed using recurrent neural networks (RNNs), which are capable of learning dependencies between sequences of images, via chains of cells with shared weights. In [16], Wang et al. combine CNNs and RNNs to achieve state-of-the-art results in multi-label image classification, exploiting semantic dependencies between the labels; this approach is relevant for our problem, in which we predict multiple diagnoses based on MRI series. CNNs are still the most popular models in medical imaging applications, though RNNs are gaining popularity, and have successfully been used for localization and segmentation tasks involving sequence-based imaging data [10]. In [8], a two-phase CNN-spatial-encoder and RNN-temporal-decoder architecture was used to identify specific (end-diastole and end-systole) frames from cardiac MRI sequences. Nevertheless, joint CNN-RNN architectures for medical image classification on time series data remain largely unexplored.

2.3. Multi-task Learning

Given the nature of the problem in predicting three separate but related tasks, we explore multitask learning [2]. In medical imaging tasks, where there is already a scarcity of data, multitask learning is useful to combine training samples for each of the related tasks [19]. Specifically, a feature learning multitask approach assumes that *m* related tasks share common feature representations. Therefore, the final classification layer of the neural network, which in the case of single-task learning would output a single class label, will now output *m* class labels [19].

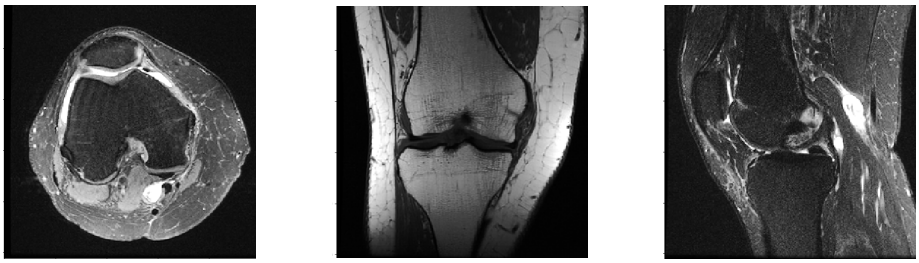
In the present work, we extend the MRNet model to a multitask approach, instead of training separate models for each task [1]. Furthermore, we explore different spatiotemporal approaches, such as joint CNN-RNN and 3D CNN architectures.

3. Dataset

Our models are trained and tested on the *MRNet* dataset [1], which consists of 1,250 knee MRI exams collected by the Stanford University Medical Center.

The exams were given binary labels across three categories: general abnormality, ACL tear, and meniscal tear; the exam labels were provided by radiologists through manual extraction from clinical reports. Most exams were ordered in response to acute or chronic pain, follow-up or preoperative evaluation, and injury/trauma. Most (80.6%) of the exams were classified as abnormal, including 23.3% of which were classified as ACL tears and 27.1% of which were classified as meniscal tears. Note that about 37% of the abnormal exams are not classified as ACL or meniscal tears, as many abnormalities reside within the cartilage and underlying bone [17]. Moreover, ACL and meniscal tears tend to co-occur, so it is relatively rare for an exam to be labeled as an ACL tear, but not as a meniscal tear.

For each exam, axial (proton density weighted), coronal (T1 weighted and T2 with fat saturation), and sagittal (proton density weighted and T2 with fat saturation) series were taken; the number of image slices per series ranged from 16 to 61. The variable number of slices per series and per exam is due to differences in the number of snapshots taken by different MRI machines across different planes of the knee. Images were scaled to 256×256 pixels and a histogram-based intensity standardization algorithm [11] was applied (so that pixels with similar values would correspond to similar tissue types). The data is divided into axial, coronal, and sagittal series; for each series, the data has dimensions $s \times 256 \times 256$, where s is the number of image slices from a single MRI exam. Figure 2 shows example images of each series type.



(a) Axial scan (b) Coronal scan (c) Sagittal scan

Figure 2: Visualization of axial, coronal, and sagittal MRI scans

3.1. Train, validation, and test splits

Since the MRNet dataset was released as part of a competition, there was no test set provided. The data we received consisted of a training set of 1,130 exams and a validation set of 120 exams. We designated 120 exams from the training set as validation data, and then used the provided validation set as our test set. Hence, our training, validation, and test sets had 1,010, 120, and 120 exams, respectively. Since the label distribution was skewed, we ensured that, when sampling from the given training set to construct our validation set, at least 50 positive examples were selected for each task. Because our training set is reduced relative to the competition training set, and since we use different validation and test sets, our results cannot be compared directly to the original *MRNet* results.

3.2. Further Preprocessing

Medical imaging tasks rely on relatively small datasets and stand to benefit from data augmentation techniques as outlined in [6]. In the present work, we augment the MRI sequences in the *MRNet* dataset by rotating each image by a randomly selected angle between $-30°$ and $30°$, thereby effectively doubling the size of our training set. In addition, we perform pixelwise standardization across all exams.

4. Methods

As the goal is to predict binary labels for each of three tasks (general abnormality, ACL tear, and meniscal tear), we make the assumption that features relevant for one task may also be relevant for another, and use a multitask learning framework. By training a single model to output a prediction for every task, we greatly reduce the computational cost compared to training a separate model per task. Within this multitask framework, we explore three methods, the latter two of which capture the sequential, 3D structure of MRI exam images: 1) Multi-MRNet, 2) CNN+LSTM and 3) 3D CNN. Due to class imbalance across the tasks, we experimented with training our models using both standard binary cross-entropy loss and weighted binary cross-entropy loss, defined as follows:

$$L = \frac{1}{N}\sum_{i,j}\frac{1}{W_j^{(i)}}\left((1-y_j^{(i)})\log\left(1-p_j^{(i)}\right) - y_j^{(i)}\log p_j^{(i)}\right)$$

where N is batch size, j represents the specific task and i a single training example; $y_j^{(i)}$ is the true label of example i for task j and $p_j^{(i)}$ is the predicted probability of example i for task j, computed by taking the sigmoid of the score output by the models. $W_j^{(i)}$ is a weight proportional to the prevalence of the ith example's class within the task j, $W_j^{(i)} = 1$ in the case of standard loss. We optimize the loss function using mini-batch gradient descent.

To avoid overfitting, we add $L2$ regularization to the cross-entropy loss objective. When using the entire exam, sequence lengths between exams may differ, and AlexNet can only support a fixed batch size. To allow for batching, we use padding or select only the middle slices from an exam.

4.1. Multi-MRNet

The original MRNet model in [1] uses a CNN to map 3D MRI sequences to probabilities. AlexNet is first used as a feature extractor to convert each MRI sequence of s images to a $s \times 256 \times 7 \times 7$ tensor. Then, a global average pooling layer (GAP) is used to reduce the dimension of the tensors to $s \times 256$, and a max-pooling layer, to obtain a 1×256 vector, removing the slice dimension. Finally, a fully-connected layer with sigmoid activation is used to compute the class probabilities. The key component of this model is the GAP layer, which reduces the number of parameters in the model by eliminating the height and width dimension of the image, thereby reducing the risk of overfitting.

The Multi-MRNet model represents a multi-class, multi-view extension of MRNet. Specifically, three MRNets, i.e. one for each view, are deprived of the last fully-connected layer and simultaneously trained with shared weights. The outputs of the three networks are then concatenated and passed through a fully-connected layer with sigmoid activation to output three probabilities, i.e. one for each task (Figure 4).

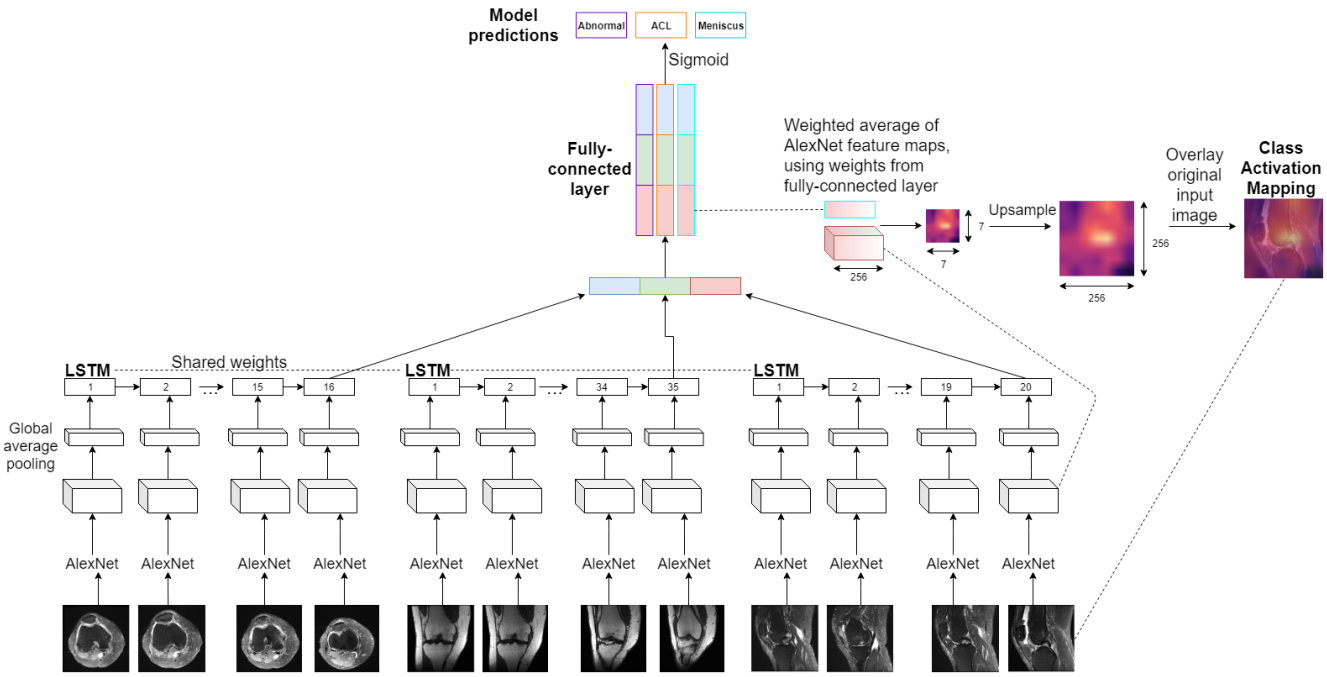


Figure 3: LEFT (Multi-view, Multi-task CNN+LSTM architecture): For each exam, the axial, coronal and sagittal image sequences are passed to AlexNet feature extractors, before being fed into a variable-length, many-to-one LSTM with shared weights across the different planes. The output from the final hidden state of each LSTM is fused and passed to a fully-connected layer with three neurons and sigmoid activation, producing probabilities for general abnormality, ACL tear, and meniscus tear. RIGHT: Procedure for producing a meniscus tear activation mapping for an example sagittal image.

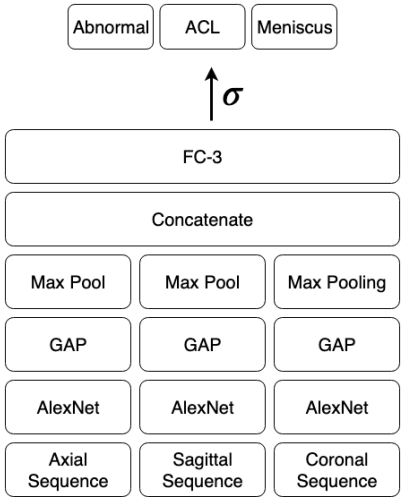


Figure 4: Multi-view, Multi-task version of MRNet

4.2. CNN+LSTM

The Multi-MRNet treats each slice of the MRI sequence as independent and outputs a different prediction for every image; the exam-level prediction is then generated by simply averaging across the batch of slice-level predictions. This does not capture conditional relationships between predictions at different timesteps. To more explicitly model the temporal dependencies across frames over the course of an exam, we add a RNN decoder on top of the original CNN architecture. For our recurrent layer, we experiment with Long Short-Term Memory (LSTM) units and Gated Recurrent Units (GRUs), both of which have been shown to alleviate problems with vanishing and exploding gradients common to vanilla RNNs [4, 3]. Then, we concatenate the recurrent layer outputs for the axial, coronal, and sagittal sequences and pass them through a fully-connected layer to generate task-level predictions (Figure 3).

4.3. 3D CNN

Similar to RNNs, 3D CNNs can also capture correlations between adjacent images within an exam sequence. In a 3D CNN, the convolution happens not only spatially across a single 2D slice, but also temporally through a sequence of slices. In the case of a MRI, this temporal dimension is a third spatial dimension, since the sequence of images in a MRI move through the depth of the body. We make the assumption that if a feature is relevant in a given MRI slice, it may be useful in an adjacent slice in the MRI series. 3D CNNs allow for such parameter sharing across the temporal dimension of a MRI series.

We began by training a multi-task 3D CNN with three

Layer	Activation Vol. Dim.
INPUT	(1,16,3,256,256)
AlexNet	(16, 256, 7, 7)
CONV3-5-p2-s1	(1, 5, 258, 9, 9)
POOL2	(1, 5, 129, 4, 4)
CONV3-3-p2-s1	(1, 3, 131, 6, 6)
POOL2	(1, 3, 65, 3, 3)
CONV3-3-p2-s1	(1, 3, 67, 5, 5)
POOL2	(1, 3, 33, 2, 2)
RESHAPE	(1, 396)
CONCAT	(1, 3 $\times$ 396)
FC-3	(1, 3)

Table 1: AlexNet + 3D CNN Layers

convolutional layers from scratch. However, given the relatively small size of the dataset and the computational cost, we decided to utilize transfer learning and train what is known as a *top-heavy* 3D network on top of the pre-trained AlexNet 2D convolutional architecture, similar to the approach in [18]. For each exam, we run each of the three sequences - axial, coronal, sagittal - through the top-heavy 3D CNN, combining outputs from the three sequences, and outputting a prediction for each of the three tasks. The layers of the model are as shown in table 1, where the layers are denoted as follows:

- CONV3-N-p2-s1 is a 3D convolutional layer with kernel size 3x3x3, N output channels, padding 2, stride 1, and ReLU activation
- POOL2 is a 3D max pooling layer with kernel size 2x2x2, stride 2, and no padding
- CONCAT is the flattening and concatenation of axial, sagittal and abnormal views
- FC-N is a fully connected layer with N neurons to output class predictions based on axial, sagittal, and abnormal views after they have been fed through the previous layers

4.4. Hyperparameters

We tune our hyperparameters on the validation set. For all of our models, we use an annealing learning rate schedule which starts at $1e-5$, decreasing by a factor of 0.3 whenever the validation loss does not improve for 5 consecutive epochs. We experimented with different initial learning rates, and $1e-5$ yields the best performance. We use the Adam optimizer to train our models, and add L2 regularization to our cross-entropy losses, with a regularization strength of 0.05. In addition to L2 regularization (and data augmentation), we also experimented with dropout and batch normalization as regularizers.The mini-batch size varies across models depending on memory constraints. For the CNN + LSTM model, we use a batch size of 8, and for the 3D CNN model we use a batch size of 10.

4.5. Class activation mappings

In general, abnormalities are localized in distinct regions of the knee. To visualize, which areas of each input image were most important in influencing model predictions, we generated weighted feature maps as follows:

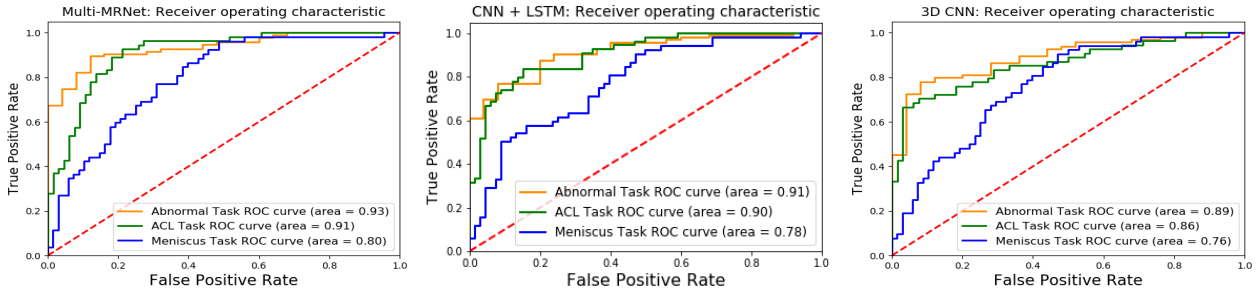
- Using parameters from the final fully-connected layer of the network, we computed a weighted average of the 256 CNN feature maps given by the AlexNet feature extractor. This linear combination of feature maps produced a single 7×7 image.
- We then mapped the 7×7 image to a heat map color scheme and upsampled to 256×256 pixels.

We overlaid these class activation mappings (CAMs) on the original (unnormalized) MRI image, allowing us to compare the model's relative weighting of different input features for a given prediction task, to the ground-truth location of the corresponding abnormality. Note that each MRI image has a different associated CAM for each of the three prediction tasks, since the final classification layer learns distinct weights for each task. While there is well-established literature on the localization properties of CAMs produced for models with convolutional layers followed by a global average pooling layer and a dense layer [20], we also experimented with producing a CAM for the CNN+LSTM model, which has these attributes, in addition to an intermediate recurrent layer. See Figure 3 for an example of how a meniscus CAM was computed for a sagittal scan.

5. Results

All the models are optimized by experimenting on the following components: 1) loss function, 2) data augmentation and 3) length of the MRI series.

Specifically, we investigate the effect of using standard or weighted cross-entropy loss and augment the MRI series using random rotations. In addition, since peripheral frames of the sequences are noisy and do not include very useful information as observed in Figure 8, we study the effect of using full-length series versus considering the middle subsequence of length L. In the case of sequences of length less than L, we either employ zero-padding or fill the sequences with random rotations, in order to keep the length of the series the same. This is especially important for the CNN+LSTM and 3D CNN models.



(a) Multi-task Multi-view MRNet (b) CNN + LSTM (c) 3D CNN

Figure 5: ROC Curves by model. The ROC curve is a plot of FPR vs. TPR for all possible decision boundary thresholds from 0 to 1, where FPR = FP/(FP+TN) and TPR = TP/(TP+FN). The AUC metric is the area under the curve.

5.1. Multi-MRNet

For the Multi-MRNet architecture, the best performance across all the experiments was surprisingly achieved by the model trained on standard cross-entropy loss with no data augmentation and full-length sequences. Nevertheless, the performance of the Multi-MRNet model is quite similar throughout all the experiments. The fact that neither weighted loss nor data augmentation improve the results is probably due both to the qualitatively different nature of the abnormality task (given that many abnormalities are neither ACL nor meniscus tears), and also due to differences in label distribution across the three tasks. While 81% of the abnormal labels in the training set are positive, only 18% of the ACL tear labels and 35% of the meniscus tear labels are positive.

5.2. CNN+LSTM

For the CNN+LSTM model, we also compare the performance of GRUs versus LSTM cells, and multi-task versus specialized models. The latter predict the "ACL tear" and "meniscus tear" tasks simultaneously, while a separate model is trained for the "general abnormality" task.

The LSTM cells consistently performed at least as well as the GRUs. In addition, the model using only the middle 6 sequences from each exam worked better than using full-length sequences. The performance of the LSTM model trained on all three tasks oscillated between favoring the abnormal task to the detriment of the other tasks, and vice-versa; in general, the model was incapable of learning a shared representation that performed well on all tasks. This might again be related to qualitative and distributional differences across tasks in the multi-task setting, as explained above. As a result, the weighted loss objective yielded much better results when the abnormal task was separated from the other two tasks.

5.3. 3D CNN

In addition to the experiments performed for the other models and general architecture, for the 3D CNN model we experimented with the following:

- fully 3D CNN from scratch vs. top-heavy 3D CNN (i.e. bottom layers 2D convolutions, top layers 3D convolutions) with transfer learning,
- multi-view approach vs. separate weights for axial, coronal, and sagittal views.

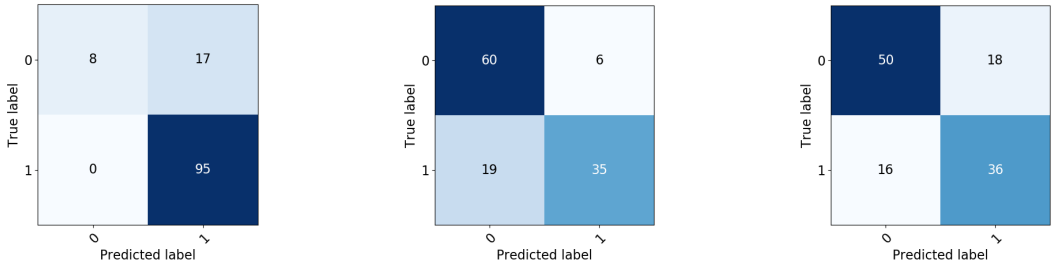
Despite batch normalization and weight decay, and experimenting with different architectures, the model trained from scratch led to heavy overfitting as noted by learning curves on the training and validation sets. As such, we focus the rest of the discussion and results on the top-heavy 3D-CNN approach.

Using separate weights for axial, coronal, and sagittal sequences outperformed the multi-view approach. The multi-view approach achieved area under the receiving operating characteristic curve (AUC) scores of 0.84, 0.82, and 0.73 respectively on the abnormal, ACL, and meniscus tasks, while using separate weights for each view improved AUC performance to 0.89, 0.86, and 0.76 on the three tasks as reported in 3. This makes sense, as the features required to diagnose a knee abnormality may appear very different in each of the views as can be seen in Figure 2.

5.4. Quantitative analysis

The quantitative comparison between different approaches is carried out in terms of their best performance over the corresponding experiments. The setups of the three models that achieved the best performance across all the experiments are summarized in Table 2:

The models must be optimized to minimize both false positives and false negatives: false positives will cause patients to undergo unnecessary surgery, while false negatives will cause patients not to receive proper care. Therefore we



(a) Abnormal Task (b) ACL Task (c) Meniscus Task

Figure 6: Best model (Multi-MRNet) performance by task. Avg. accuracy: 0.79, Avg. precision: 0.84, Avg. recall: 0.78

Model	Multi-MRNet	MRNet+LSTM	3D CNN
Loss	standard	weighted	weighted
Augm.	none	none	rotation
Seq. leng.	full	16	16
Transfer	yes	yes	yes
Mul. task	yes	partial	yes
Mul. view	yes	yes	yes

Table 2: Best models setups across the experiments.

Model	Multi-MRNet	MRNet+LSTM	3D CNN
abnorm.	0.9335	0.9149	0.8905
acl	0.9094	0.9026	0.8603
menisc.	0.7981	0.7769	0.7639
avg.	0.8803	0.8648	0.8382

Table 3: AUC scores

evaluate our models on the area under the receiver operating characteristic curve (AUC). The AUC metric is particularly useful as it is robust to class imbalance. For example, where the class labels are mostly 1 as in the abnormal task, a classifier could achieve high accuracy by always outputting 1, while its AUC score would be 0.5. In this way, AUC aids in the optimization of the rate of false positives compared to true positives considering all possible thresholds.

Figure 5 shows the receiver operating characteristic curves (ROC) for the three methods that gave the best performance across all the experiments performed. All the three methods seem to perform better in detecting the general and ACL abnormalities, compared to meniscal tear. The worse performance in the meniscus task is consistent with the MRNet results [1]. Given that an ACL tear is usually more severe than a torn mensicus, the features of a torn mensiscus may be more subtle and may require a model with much higher representational power.

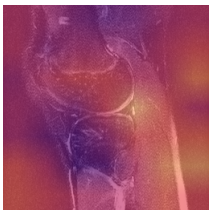
Table 3 summarizes the performance of the methods in the three tasks. Even though the performance of the three methods are quite similar, Multi-MRNet achieves the best AUC in all the tasks. This may be due to the fact that Multi-MRNet has fewer parameters, and despite regularization efforts and data augmentation, the large number of additional parameters in the LSTM and 3D CNN architectures caused overfitting. The GAP layer in the Multi-MRNet model is a key factor in reducing the number of parameters and may contribute to its success. Given its success, for the remainder of the paper we will show results for Multi-MRNet.

Figure 6 shows the confusion matrices, i.e. number of true/false positives and negatives, given by the Multi-MRNet model for the three tasks at hand. As a consequence of the high number of positive examples for the abnormal task in the dataset, the model outputs several false positives but no false negatives. This means that some patients would receive treatment even though no abnormality is actually present. Concerning the ACL task, the dataset is imbalanced in the opposite sense, namely more negative examples are present; this causes the model to output more false negatives than false positives. The class imbalance in the dataset seems to have lower effect on the meniscus task, which nonetheless manifests the lowest accuracy. In this case the number of false positives and negatives are comparable.

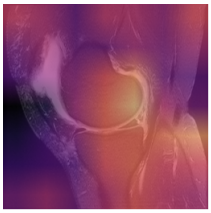
5.5. Qualitative analysis

5.5.1 CAMs

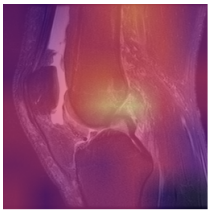
To better understand the behavior of the models, we generate class activation mappings (CAMs), where the brightest areas of the CAMs indicate the regions that most influence the model's prediction. Figure 7 shows CAMs from the Multi-MRNet and CNN+LSTM models. The CAMs from the Multi-MRNet model are interestingly more diffuse than expected, though it does seem that there is some saliency right around the meniscus. There is also a fair amount of activation elsewhere, which may stem from the multi-task nature of this model. By multi-tasking, the model likely cannot focus as much attention on fine-grained details. The CNN+LSTM CAM in Figure 7b, produced from the dual-



(a) Multi-MRNet model: Sagittal view of meniscus CAM

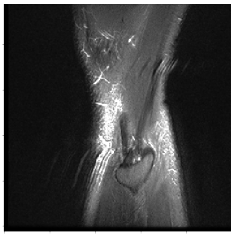


(b) CNN+LSTM model: Sagittal view of meniscus CAM

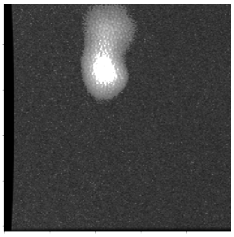


(c) CNN+LSTM model: Sagittal view of abnormal CAM

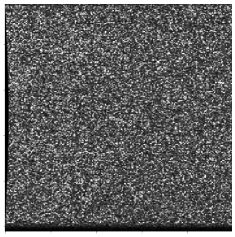
Figure 7: Class activation mappings (CAMs)



(a) A scan with artifacts



(b) An outer scan



(c) A scan which is entirely noise

Figure 8: Visualization of challenging scans

task ACL/meniscus model, is somewhat less diffuse, which may be explained by the removal of the abnormal task from the joint objective, and the CAM in Figure 7c, produced from a single-task model, has the most localized saliency. However, given the unique architecture of the CNN+LSTM model, it is unclear whether this CAM is directly interpretable. Without confirmation from a radiologist, we cannot comment extensively or draw further conclusions on these CAMs, other than to be somewhat skeptical of the activations on areas outside of the knee.

5.5.2 Problematic MRI Sequences

We discovered a handful of images which would be difficult even for a trained radiologist to read, examples of which can be found in Figure 8. For instance, some images include artifacts (perhaps from movement), while outer scans do not seem to provide any meaningful information, and there are even scans which are completely noise. While these images do not make up the majority of the dataset, they clearly provide a challenge to any human or deep learning algorithm.

6. Conclusions and future work

This paper has presented deep learning methods to detect knee abnormalities from MRI sequences. Three different methods have been investiated and tested, based on the MRNet architecture developed in [1].

First, we extended the MRNet method to process all the views in the sequences, i.e. axial, coronal and sagittal, and to output the three tasks labels, i.e. abnormal, acl and meniscal tear, simultaneously. The resulting model has been referred as Multi-MRNet. Then, we experimented with substantially different approaches, such as using RNNs and 3D CNNs to take advantage of the three-dimensional nature of the MRI sequences.

We optimized our models by experimenting over different aspects of deep learning methods, namely loss function, data augmentation and selection, activation functions, transfer learning, etc. From the numerous experiments, we selected the models that had highest performance on the test set for the comparison and discussion of the results. Specifically, standard cross-entropy loss with no data augmentation and full-length sequences provided the best results for the Multi-MRNet approach, while weighted cross-entropy loss with no data augmentation and reduced-length sequences achieved the best results for the CNN+LSTM model, and weighted cross-entropy loss with random rotations and reduced-length sequences achieved the best results for the 3D CNN. Overall, the Multi-MRNet model perfomed slightly better than the other methods in all the tasks at hand, achieving an average AUC of 0.88.

In addition to the quantitative metrics, we generated CAMs to get a better sense of the sensitivity of the models to the image features. The CAMs produced by the Multi-MRNet model seemed quite diffuse throughout the image, without focusing exclusively on meaningful areas. The reason might be related to the nature of the multi-task model, in which the model cannot specialize in detecting unique features of the image. Future work will focus on confirming with expert radiologists if the results are meaningful and the models could be employed in everyday practice.

7. Acknowledgements

The authors would like to thank Nicholas Bien, author of the MRNet paper, for the help and support throughout the project. We would also like to thank Radhika Tibrewala, a researcher and knee MRI expert at UCSF, for her advice throughout the project.

All code is implemented in PyTorch [13], building off of the code implemented by Nicholas Bien and team in [1].

8. Contributions

Giacomo worked on the Multi-MRNet model, Mara worked on the CNN+LSTM model, and Sarah worked on the 3D-CNN model. All the remaining work was shared evenly across team members. Code for this project can be found at: `https://github.com/giacomolamberti90/CS231N_project`

References

[1] 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.

[2] R. Caruana. Multitask learning. *Machine Learning, 28, 41-75*, 1997.

[3] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[4] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[5] E. Hosseini-Asl1, G. Gimelfarb, and A. El-Baz. Alzheimer's disease diagnostics by a deeply supervised adaptable 3d convolutional network. *arXiv preprint arXiv:1607.00556v1*, 2016.

[6] Z. Hussain, F. Gimenez, D. Yi, and D. Rubin. Differential data augmentation techniques for medical imaging classification tasks. *AMIA*, 2018.

[7] M. Kazemi, Y. Dabiri, and L. Li. Recent advances in computational mechanics of the human knee joint. *Computational and Mathematival Methods in Medicine*, 2013.

[8] B. Kong, Y. Zhan, M. Shin, T. Denny, and S. Zhang. Recognizing end-diastole and end-systole frames via deep temporal regression network. In *International conference on medical image computing and computer-assisted intervention*, pages 264–272. Springer, 2016.

[9] 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.

[10] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. Van Der Laak, B. Van Ginneken, and C. I. Sánchez. A survey on deep learning in medical image analysis. *Medical image analysis*, 42:60–88, 2017.

[11] L. Nyl and J. Udupa. On standardizing the mr image intensity scale. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 42:1072–1081, 1999.

[12] E. H. Oei, J. J. Nikken, A. C. Verstijnen, A. Z. Ginai, and M. Myriam Hunink. Mr imaging of the menisci and cruciate ligaments: a systematic review. *Radiology*, 226(3):837–848, 2003.

[13] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in PyTorch. In *NIPS Autodiff Workshop*, 2017.

[14] A. Payan and G. Montana. Predicting alzheimers disease: a neuroimaging study with 3d convolutional neural networks. *arXiv preprint arXiv:1502.02506v1*, 2015.

[15] V. Pedoia, B. Norman, S. N. Mehany, M. D. Bucknor, T. M. Link, and S. Majumdar. 3d convolutional neural networks for detection and severity staging of meniscus and pfj cartilage morphological degenerative changes in osteoarthritis and anterior cruciate ligament subjects. *PMC*, 2019.

[16] J. Wang, Y. Yang, J. Mao, Z. Huang, C. Huang, and W. Xu. Cnn-rnn: A unified framework for multi-label image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2285–2294, 2016.

[17] W. Wiggins. A radiologists exploration of the stanford ml groups mrnet data, April 2019.

[18] S. Xie, J. Huang, Z. Tu, and K. Murphy. Rethinking spatiotemporal feature learning:speed-accuracy trade-offs in video classification. *arXiv:1712.04851v2*, 2018.

[19] Y. Zhang and Q. Yang. A survey on multi-task learning. *arXiv:1707.08114v2*, 2018.

[20] B. Zhou, A. Khosla, A. Lapedriza, and A. T. Aude Oliva. Learning deep features for discriminative localization. *Computer Vision Foundation*, 2016.