Final Implementation Plan

Group Members: Kexin Yang, Hainan Xiong, Haoming Chen

Mentor: Alex Chowdhury

Note: Our implementation plan is hypothetical, which is based on the deep Reinforcement Learning model we built for the 3-dimension organ localization task (we started with liver localization). In this implementation plan, we will focus on the liver cancer diagnosis process at Dana Farber Cancer Institution, and propose a hypothetical implementation plan to deploy our model into the liver cancer diagnosis process. Our goal will be to provide radiologists with an easy-to-use tool that can help them process the patient's 3D CT scan and crop it to the liver region so that they can evaluate the scan more easily.

Workflow Map

1) A workflow map describing the current process that pertains to your data science solution, identifying who is doing what, as well as any other information you consider relevant for each step (e.g. work time, chronological time, effort, cost, etc.). Also, identify decision points within the process.

Our Project:

Our project implementation plan is hypothetical and will focus on the diagnosis of liver cancer. Our model will be deployed in Dana Farber Cancer Institution (DFCI) to allow the radiologist to localize the liver organ in the 3D Abdomen CT scan more easily using our model API.

Specifically, our model will output a bounding box, which is the smallest 3-dimensional box containing the liver organ, and it will crop the original CT scan to only include the liver region so that the radiologist can evaluate the CT scan more easily without having to manually locate the liver region as before. Therefore, deploying our model in DFCI can improve radiologists' working efficiency compared to their traditional manual localization method.

The figure below shows the workflow map for the liver cancer disease diagnosis process in DFCI with our model deployed in the hospital system. The rhombus-shaped boxes indicated decision points in the workflow.

Note that this workflow map only shows the scenario when the Primary Care Provider (PCP) decides that the patient needs to get an abdomen CT scan for their liver cancer diagnosis since this is the only scenario in which our model will be utilized. When a patient comes to DFCI with symptoms related to liver cancer, the PCP will determine whether an abdomen CT scan is required. If so, the patient will get the CT scan, which will be pre-processed with the current standards used in DFCI and stored in the DFCI data lake in NIFTI file format. When the radiologist pulls this patient's CT scan, they can call the *localize()* function provided by our model API, which will feed the CT scan to our model to output a liver bounding box and

generate a cropped liver CT scan. If the radiologists review the cropped CT scan and think it is accurate, they can proceed to write the report and send the PCP to support their liver cancer diagnosis. Otherwise, the radiologist will need to manually locate the liver in the original CT scan.

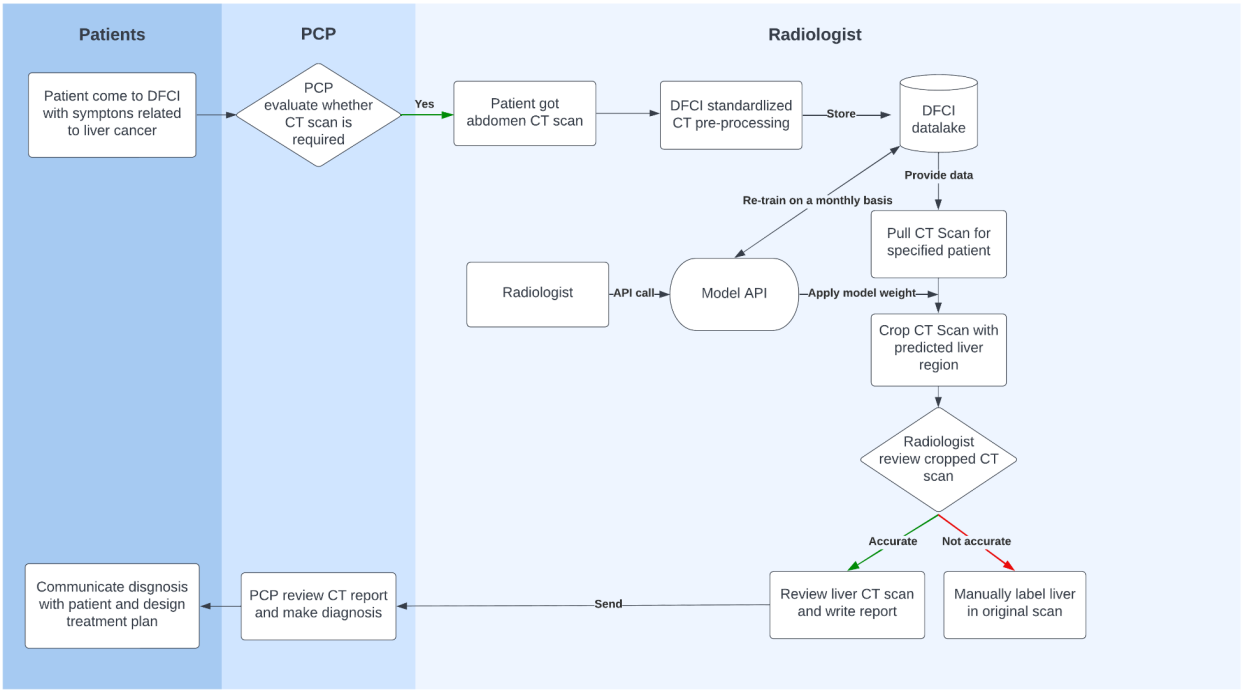

Figure 1. Current Workflow Map

2) A workflow map of the future state that includes your data science solution. Include any relevant information as well as any risks or dependencies (e.g. your workflow map may be reliant on your data science solution achieving at least a certain level of accuracy). This should reflect your initial thoughts with the understanding you can gather, this workflow map may change between now and your final report at the end of the semester as you learn more. Potentially, you can make more than one "future state" workflow map. Make sure to include the specific decisions that the model used will make based (in part) on your model output, and what will happen afterward.

Our Project:

We plan to implement two additional features in the future workflow, which include (a) a feedback report system (b) adding an inaccurately localized liver CT scan that's manually labeled by the radiologists as a new sample into our training dataset and schedule monthly re-training to continuously improve model performance. These additional features are highlighted with gray shades in the figure below.

First, the feedback report system will pop up as a new window every time the radiologist calls our model API to generate a CT scan cropped to the liver region. It will ask the question that

"How accurate is the model output for the liver location in a CT scan?", and the radiologist will be able to choose from any one of the three possible options listed below:

- Accurate and does not require additional adjustment
- Somewhat accurate and require a little adjustment
- Not accurate and need to locate organ by themselves

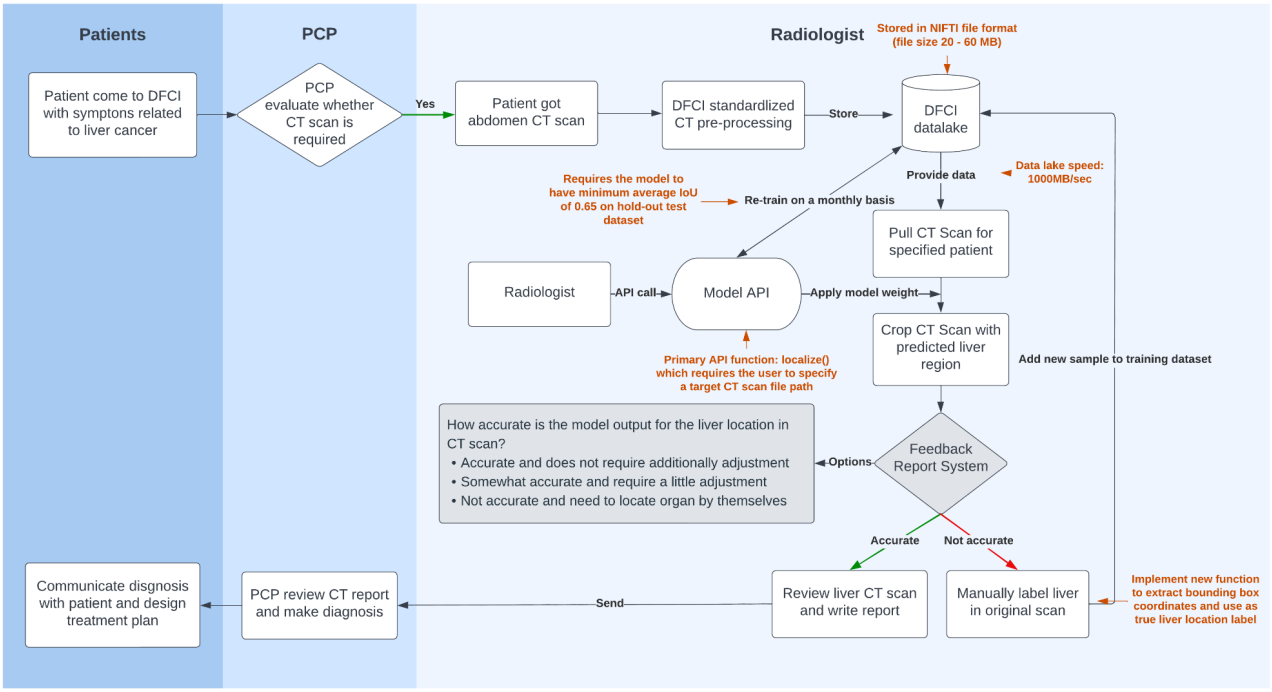

Figure 2. Future Workflow Map

The radiologists can simply click the button next to the option to indicate their choice and submit the answer. We will record the answer and monitor the model accuracy reflected by the answer to make sure the model can maintain a good performance. If a significant decrease in model performance is detected through this feedback report system, then we can quickly notice it and take action to analyze the source of the problem.

Second, whenever the radiologist decides that the cropped liver CT scan provided by our model is not accurate enough, they will be able to manually adjust the scan region. For our future workflow, we will implement a new function to record the bounding box coordinate of the CT scan region after being adjusted by the radiologists. This CT scan and the recorded bounding box coordinate will be added to our DFCI data lake as a new training sample. We will also schedule the model to be re-trained on the new training dataset on another DFCI server and test its performance on a hold-out test dataset. If the model shows improved performance after

re-training, then the updated weights will be copied into the server where our model weight is stored and used to replace the previous weights. This way we can make sure that our model performance is continuously improving.

Modeling Plan

1) Context about how your model will be used since this will affect some of the modeling decisions.

Conventional manual localization is time-consuming, and we want our deep reinforcement learning model to be able to output a bounding box (the smallest 3-dimensional box containing the liver organ) for the liver organ to help the radiologist evaluate the scans more easily. We plan to deploy our model in DFCI's system and implement a model API. Radiologists in DFCI can call the model API, and provide the 3D CT scan that they need to evaluate, and our model can output the liver bounding box and automatically crop the 3D CT scan to only include the bounding box region. The radiologist can then simply review this cropped liver CT scan and evaluate the patient's condition, which can reduce their evaluation time. In the case, they think the cropped liver scan is not accurate, a manual option to adjust the CT scan display region will also be provided.

The model will be pre-trained before being deployed, and hypothetically the model will have a high liver localization IoU score (IoU score > 0.65) and good generalizability before the deployment. It will be trained on existing 3D CT scans data points from DFCI, where each 3D CT scan will be the game environment, the voxel values inside the bounding box will represent the states, and an agent will be trained to take actions (chosen from 11 possible actions) to move or transform the bounding box to locate the liver in the CT scan.

2) Discussion about which accuracy metrics you will focus on and why you believe these are the appropriate accuracy metrics for your use case.

We will mainly be focusing on the Intersection over Union (IoU) metric since this is the most commonly used metric to evaluate image segmentation and localization tasks. It can give an overall evaluation of how accurate our predicted localization box is compared with the true label box. The IoU score is calculated as below:

$$IoU = \frac{Area\ of\ overlap}{Area\ of\ Union}$$

Since our baseline model also used IoU score as their accuracy metric, this can also allow us to directly compare our model performance with our baseline.

3) You modeling plan, including which model(s) you will try, in what order you plan to try them, and why those models in the context of the data you have (size and type of data), as well as the future use of your model.

Baseline:

We will directly use the results from the paper *Deep Reinforcement Learning for Organ Localization in CT* (Navarro et al., 2020) as our baseline since our model is directly built based on their paper. We will not re-train the baseline on the dataset we are using due to both technical and time limitations. The baseline paper did not publish their code and only provide very limited information about the hyperparameter they used in the paper, therefore it is impossible to completely replicate their model on our dataset. However, since the dataset they used are very similar to our dataset, we think it is reasonable to directly compare the results between our model and the baseline.

Our Model:

We proposed 3 novel approaches to improve the baseline model: (a) an optimized reward function; (b) a new termination condition; (c) a novel action update method. These three approaches together are designed to both reduce computation time and increase accuracy. We will implement our model and train it on our dataset.

There are 2 main reasons for choosing to use a deep Reinforcement Learning model for 3D organ localization tasks. First, most state-of-the-art (SOTA) deep Convolutional Neural Network (CNN) models designed for medical imaging such as U-Net are computationally expensive. This will results in a long inference time due to the large model size, and in our use case in DFCI, the radiologists may need to wait for a long time before the model output the bounding box for liver location. This directly violates our goal of reducing radiologists' time for CT scan evaluation. On the other hand, our baseline paper has shown that deep RL models have the potential to achieve equivalent performance with much smaller model size, so the inference time will also be very short. Second, most of the deep CNN models require a large corpus of annotated training data in order to achieve a good performance, making them prohibitive for medical tasks since it is hard to obtain a sufficient amount of annotated data, especially for 3D CT scans. Deep RL models are able to obtain good performance with limited training datasets.

If our model can achieve an equivalent or better performance on the CT dataset compared to the SOTA models, then we will deploy our model in the DFCI system and implement a model API to allow the radiologist to localize the liver in the CT scans more easily.

4) Discussion about "nice to have" and "absolute minimum" accuracy (based on the metrics you defined) given your use case, as well as contingency plans if the accuracy is lower.

	Avg IoU	Wall dist [mm]	Centroid dist [mm]
Right Lung	0.77	3.46 ± 5.28	6.06 ± 10.25
Left Lung	0.73	4.91 ± 7.38	10.32 ± 17.09
Right Kidney	0.60	2.96 ± 2.91	5.69 ± 5.67
Left Kidney	0.57	4.06 ± 4.98	7.52 ± 9.02
Liver	0.80	2.41 ± 0.70	3.36 ± 1.34
Spleen	0.60	5.25 ± 7.23	9.20 ± 12.03
Pancreas	0.32	12.26 ± 13.60	20.79 ± 20.38
Global	0.63	5.04 ± 6.01	8.99 ± 10.82
Median	0.60	2.25	3.65

Table 1. Average IoU score for different organ localizations obtained by the baseline model

The table above shows the average IoU score for different organs from the baseline paper. We can see that the baseline model achieves a 0.80 IoU score for liver localization. This will be used as the “nice to have” model accuracy since the baseline model is trained with a larger dataset and therefore is expected to have better performance.

We will set the 0.65 average IoU score as our “absolute minimum” accuracy. This is because when we examine the bounding box in a 3D CT scan and compare the prediction with the true label, we find that the bounding box with an IoU score > 0.65 already covers a large liver region. Therefore, in our actual deployment, when cropping the original CT scan based on the predicted bounding box, we can allow a slightly larger region to be included. This way, we can make sure that the entire liver region is included in the cropped CT scan most of the time.

In the case of lower accuracy, we will continue to train the model to try to achieve the “absolute minimum” accuracy of 0.65 before deploying the model. This is because if the model cannot obtain a good liver localization accuracy, then deploying the model in DFCI will not be useful for the radiologist since the cropped CT scan is inaccurate most of the time. Doing so will not help

reduce the time for radiologists to evaluate the CT scan, and may even increase their workload. Therefore several ways to improve the model's localization accuracy: (a) increase training epoch and conduct more hyperparameter tuning. (b) use a larger training dataset and increase the amount of training samples (c) optimize the model architecture.

Model Deployment and Maintenance Plan

Model Deployment Plan

How you plan to deploy your model given what you know about your user institution's infrastructure: how the data will be pulled, how it will be pre-processed, how the model will run to create outputs, how those outputs will be communicated to the system that will host the user interface.

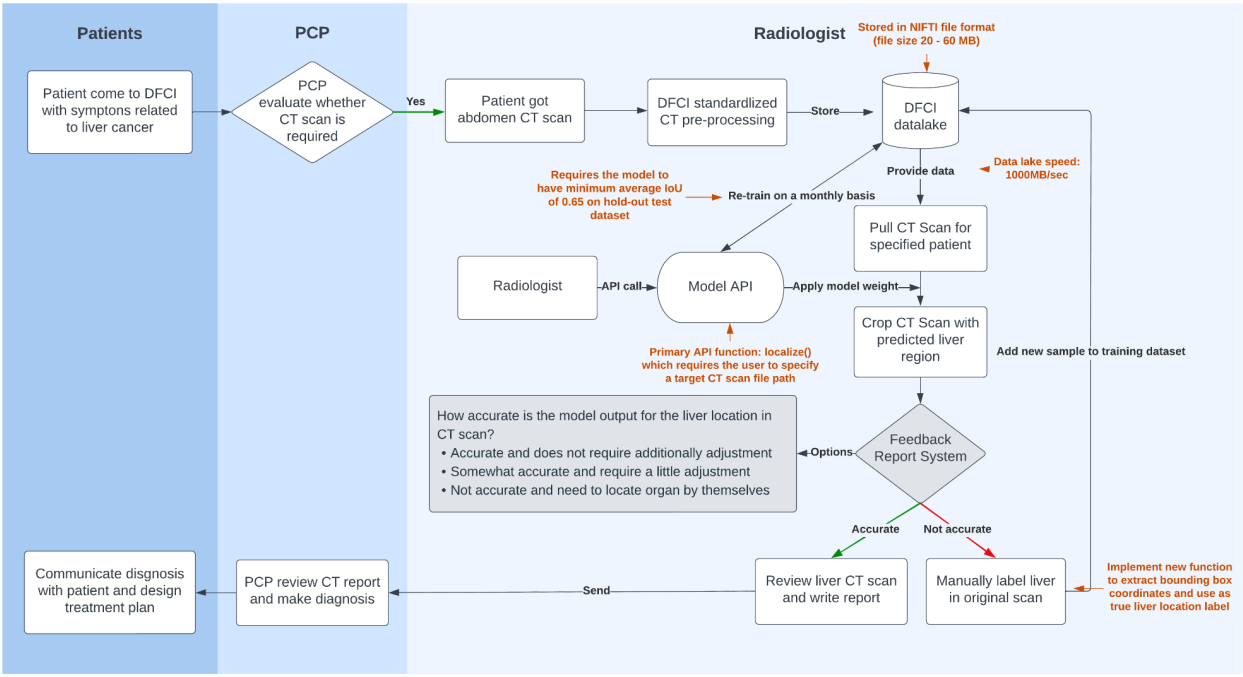

Figure 3. Future Workflow Map

Our model will be deployed on DFCI as an assistant tool as a part of the disease (liver cancer) diagnosis process to help the radiologist improve their working efficiency when evaluating the abdomen CT scans by automatically localizing the liver area. The specific use case procedure and model deployment plan are described below:

1. After a patient came to DFCI and has their abdomen CT scanned, the 3-dimensional CT image data will be stored in an existing DFCI/MGB data lake after going through the standardized CT pre-processing steps that DFCI is using right now. So in the data lake, patient information will be stored along with the processed CT scan. We plan to store the pre-processed CT scan file in the NIFTI file format, which is a common format to store medical imaging data, and we expect the file size to range from 20 to 60 MB, which is a common range for 3D CT scan stored in this format. Since the file size is not very big, we do not expect the data lake to have speed issues. On the most common infrastructure, one PC can read these files at a speed above 1000MB/sec.

2. We will provide API functions that radiologists can call on the DFCI server to pull the CT images from the data lake by specifying the patient information such as the patient's MRN. We will provide multiple training sessions to provide a detailed introduction to how to use the API functions before the model is put into production at DFCI. We will also provide assistance to continue to help them with any usage issues after the model is used to make ensure our users won't be stuck with technical problems. The primary function in the API will be a function called *localize()* which requires the user to specify a target CT scan file path, as well as some optional parameters such as the windowing range to pre-process the CT scan, and the *localize()* will feed the CT scan into our model, obtain the liver bounding box and return the cropped CT scan to the user.

3. The radiologist can then call our model on demand via another API function to obtain a CT scan that is cropped to only include the liver region as predicted by our model. We will also create a feedback report system where the radiologist will be asked to review the cropped liver CT scan and evaluate whether it is accurate or not. If not accurate, then the radiologist will be directed to another system where the original CT scan is provided, and they need to manually locate the liver region, and the system will record the correct bounding box coordinate created by the radiologist and add this sample back to the training dataset. The model will be trained on the updated dataset on a regular basis (weekly or monthly). The feedback report system will be implemented as a survey in a separate window which will pop up after the radiologist finishes evaluating the CT scan. There will be three options to score the CT scan accuracy:

   How accurate is the model output for the liver location in the CT scan?

   - Accurate and does not require additional adjustment

- Somewhat accurate and requires a little adjustment
- Not accurate and needs to locate organ by themselves

The radiologists can simply click the button next to the option to indicate their choice and submit the answer.

4. The radiologist will review the correct liver CT scan generate, write a report, and send it to PCP through the hospital communication system, and the PCP will get the CT scan and report, evaluate it with other information and give a diagnosis.

Model Maintenance Plan

A realistic plan for model maintenance given the use case. Consider how costly this model maintenance will be, as well as what the requirements are for the model (how critical is your model? how long can it be down? how bad is it if its accuracy fails?)

We think there are mainly two possible issues that our model may encounter when deployed in reality:

Alarm fatigue

This may occur if the model output is consistently inaccurate and the radiologist simply ignores the model output and look at the original CT scan and locate the Organ of interest (liver) by themselves. This issue can be solved relatively easily by making sure the model is trained to have good accuracy, which is realistic since our baseline model is already able to achieve a high IoU score when evaluated on the testing dataset. We will also continuously add more samples created by the radiologist back to our training dataset, as described in step 3 in the previous question where the radiologist will be asked to manually identify the liver region in the original CT scan if they think the liver localization generated by our model is inaccurate. We will identify the bounding box coordinates of the liver identified by the radiologist manually and add these samples to our training dataset. This way we can make sure the model can continue to learn with a larger and larger dataset and make better predictions. The re-training will be scheduled to occur on DFCI on a separate server monthly, and the updated weight will be applied if it can achieve a better IoU score on the hold-out testing dataset.

Algorithm complacency

This may occur if the model prediction is consistently accurate, and the radiologist will automatically accept the model output without any careful evaluation. In our use case, the radiologist is still required to review the cropped liver CT scan and write a report to the PCP, therefore we think algorithm complacency will not cause a big issue. However, we will implement a mandatory feedback report system that requires the radiologist to judge the accuracy of the model. We will also continuously add new labeled datasets to make sure the model accuracy can be maintained.

The model will be maintained by continuously monitoring its performance and collecting new training samples through the feedback report system, and regularly training the model to maintain its prediction accuracy. Overall we think the maintenance cost is relatively low since once the system is deployed, it will be able to run and update automatically.

The critical level of our model is low since the radiologists need to review the CT scan in order to write the report, so they will always be able to capture inaccurate model prediction, and our system will allow them to view the original scan and manually locate the liver. Therefore it will not affect the normal disease diagnosis process if our model is down or its accuracy failed except the radiologists will take a longer time to review the CT scan.

a) How you will troubleshoot your model if it fails. What are some possible failure modes and how they can be fixed.

We think possible failure modes include:

- Drift in data quality
- Drift in feature attribution
- Bias in model prediction

In our case, these drifts may occur if the hospital makes some changes to the protocol of the CT scan or the standardized pre-processing methods. This may cause the new CT scan will have different parameter settings such as volume, in-plane resolution, the field of view, etc, and the model may not be able to make accurate predictions due to these drifts.

To account for these problems, we plan to use a mix of online and offline retraining to fix the model. The key is to monitor for these drifts once they occur, and we can retrain the model in a scheduled manner and complement it with real-time training when there is a sudden change in model performance. The specific methods of monitoring are described in the next part.

b) How you will detect and deal with direct changes in your input data (e.g. changes in the structure)

We will implement 2 methods to monitor for drift in data quality and distribution:

Before prediction

Prior to the prediction, we can simply monitor the input distribution, the label distribution, or the conditional distributions of features given label data to detect data shift. specifically, we can directly compare their statistical property, such as mean, median, and variance of features including the voxel volume and average pixel intensity of the liver. We expect most of the CT scan image and label metadata to be in the same NIFTI file format. For metadata that is provided in another format, we will try to convert it into NIFTI file format. In the case where point estimates are insufficient to detect the data shift, we can use KL divergence or Wasserstein Distance to measure the difference between two distributions.

After prediction

We will use the sliding window method to evaluate the radiologist's feedback on the model prediction accuracy. Using the sliding window method to monitor trends and outliners in time-series data is described in the paper by Yu et al. (2014). We can monitor for data drift by applying a sliding window on past feedback and see if there is a continuous decrease in the feedback score. An advantage of using the sliding window method is that it can help us distinguish an overall data drift versus an individual outliner.

c) How you will detect and deal with practice changes that may render your model obsolete.

Communicate with Radiologists regularly

We will communicate with radiologists in DFCI that use our model API on a monthly basis to collect their overall feedback about the model accuracy on liver localization and see if they detect any obvious decrease in model performance on their daily use. If so, we will conduct more thorough research on the model performance to identify the reason for its decreased performance.

Monitor model performance

We will build a dashboard for the users where they can view exportable reports and graphs for the model's performance and configure alerts to receive notifications if it is detected that the model's performance declines beyond a certain threshold.

Continuous update dataset and model training

We will continuously add new samples to the training dataset by asking the radiologist to manually locate the liver in the CT scan and generate the correct bounding box coordinate whenever the model prediction is inaccurate. We can account for possible data drift using this method since new data will be added and the model will be trained on the dataset that represents the "drifted" distribution.

d) How you will check your model accuracy to test for model drift.

We will take a sliding-window approach to continuously monitor the model accuracy and determine whether the model drift is temporary (due to some short-lived events) or permanent. For example, we can set the frequency of the checks to 2 days. This means that we compute the model accuracy on data collected during a 2-day window. An alert is issued if the value falls outside of an allowed range A. We will use the radiologist's label as ground truth and check how much the model prediction disagrees with the radiologist.

Study and Change Management Plan

Study Plan

"how will you measure whether your tool is helpful?"

Since the primary motivation for our project is to help radiologists improve their working efficiency in the disease diagnosis process, the primary measure for our tool's helpfulness should be the radiologist's feedback, which we will collect through a feedback report system.

Additionally, we will also use other metrics, such as the latest evaluation IoU score (which will be updated on a monthly basis) and the radiologist's average evaluation time spent on each CT scan before and after the tool is implemented, to provide a more objective measurement on its helpfulness.

1) What will you measure? What will be your primary metric? What are other things you want to measure?

Primary Measure: Radiologist's feedback

We will implement a feedback report system that requires the radiologist to report the accuracy of the model output for each CT scan to measure its accuracy on the organ localization task and helpfulness for the radiologist's job. The radiologist will be asked to choose from three possible options, as listed below:

- Accurate and does not require additional adjustment on organ region
- Somewhat accurate and require a little adjustment on organ region
- Not accurate and need to locate organ by themselves

These answer will be collected for each instance of CT scan organ location prediction from all radiologist that uses this tool. The results will be gathered together and used as the primary measure for the tool's usefulness. If for the majority of the time, the majority of radiologists think the model output is accurate or somewhat accurate, then this would support that the tool is useful, and vice versa.

Additional measurement 1: Evaluation IoU score

Whenever the radiologist thinks the model output is not accurate enough and made a manual adjustment on the organ location bounding box using the provided toolbox, our system will record this bounding box location as the true label for the CT scan and add it to the training dataset. Therefore, our training dataset will be updated, we will re-train our model on a monthly basis, and the IoU score on the hold-out evaluation dataset will also be used as a metric to evaluate the tool's usefulness. If the evaluation IoU score is high and/or has an increasing trend, then this will be objective evidence to support its usefulness.

Additional measurement 2: Pre-post average evaluation time

The average amount of time that each radiologist spends to evaluate the CT scan will also be an important metric to measure the tool's usefulness. Therefore, we will record the amount of time each radiologist spends to evaluate the CT scan for a week before the tool is implemented and calculate the average to establish a baseline for each radiologist. After the tool is implemented, we will continue to record the time spend on each CT scan and track the running weekly average for each radiologist. As long as the post-average evaluation time is shorter than the pre-average evaluation time, this will support that our tool is useful. As the radiologist becomes more familiar with the system and the model achieves higher accuracy with more training datasets, we would even expect to see a continuously decreasing trend in the weekly average evaluation time.

Additional measurement 3: Pre-post health outcomes (monthly disease diagnosis rate)

We will also measure the patient's health outcome, specifically the monthly disease diagnosis rate, to account for potential problems with the tool. We would not expect the monthly disease diagnosis rate to be vastly different before and after the tool is implemented. However, if there is a significant difference (ex: a much lower diagnosis rate), then we will need to conduct additional investigation to see if this is valid, or if there is any problem with the tool. For instance, the model returned inaccurate organ location but the radiologists failed to detect this

error and did not report the disease symptoms/signs from the CT scan. If this happens, then improvements will be required before the tool can be used again.

2) What will be your study design? What's your comparison group? Why is this study design feasible? How many data points do you expect to have?

This will be primarily a pre-post study that compares the radiologist's efficiency in evaluating CT scans before and after the tool is implemented. Both subjective and objective metrics will be recorded to evaluate the tool's usefulness. The comparison group will be the metrics (average evaluation time, monthly disease diagnosis rate) recorded prior to the tool implementation. This is feasible since we can easily implement a toolbox to record the time radiologists spent to evaluate each CT scan, and calculate the monthly disease diagnosis rate from the hospital's EHR. We would expect to have data points of all such CT scan evaluations within a month prior to the tool implementation.

3) Is your study research or quality improvement? Why? If it's research, who are your subjects? How will you gather subject consent?

Our study is for quality improvement. This is because the goal of our study is simply to improve the radiologist's working efficiency by helping them locate the organ more quickly. We do not conduct research to investigate an optimized method for disease diagnosis, and we don't expect the tool to improve the disease diagnosis accuracy either. The model output is simply provided as a suggestion to the radiologists, and they have the option to neglect this suggestion and apply the traditional method of locating the organ by themselves.

Change Management Plan

How will you ensure your users and stakeholders are on board?

1) Who are your primary users? Who are indirect users? (other people who may not directly use the tool but who may have their workflow affected as a result). Who are other stakeholders? (e.g. leadership, etc.)

<u>Primary users: Radiologists</u>

Radiologists are the primary user of the tool since they will directly use the tool in their job to evaluate the CT scan image.

<u>Indirect users: Primary Care Provider</u>

Primary Care Providers are the indirect user of this tool since the PCP will receive the CT scan report from radiologists who uses the tool, and make the final disease diagnosis decision based on the report (along with other materials).

<u>Stakeholders: Hospital leadership</u>

are the stakeholder since the deployment of this tool in the hospital can may improve the working efficiency of the radiologist, which is beneficial for the hospital. However, it is important

for the leadership team to monitor the performance of this tool to see whether it achieves this objective in reality, and take the appropriate reaction if the implementation of this tool yields any negative results.

2) Show a list of stakeholders, indicating who you think will be primary users, indirect users, and others. Describe your plan of when you will meet with each of the groups, and how often. For your primary and indirect users, plan the number of meetings for each element of the ADKAR framework, and roughly when they will happen. Give timeframes.

The graph below shows the timeframe for each stage in the ADKAR framework, as well as the number of planned meetings with the primary user, indirect user, and stakeholder.

Primary user

We plan to meet with the primary user (radiologists) on a weekly basis during the pre-intervention period to share information about the tool, such as its functionality and how to use it, etc, and we will collect feedback from the radiologists after each meeting to learn about their suggestions and see if we can incorporate the good suggestions into the tool. Then during the post-intervention period, we will gradually decrease the meeting frequency and hold the meeting on a bi-weekly and then monthly basis. The meeting will likely happen on a fixed schedule at the beginning of each week.

Indirect user & Stakeholder

We plan to meet with the indirect user (PCP) and stakeholder (hospital leadership) once at each stage in the ADKAR framework to keep them updated about the tool's implementation and functionality. They will be invited to join the first meeting at each stage. We will also collect feedback from them after the meeting.

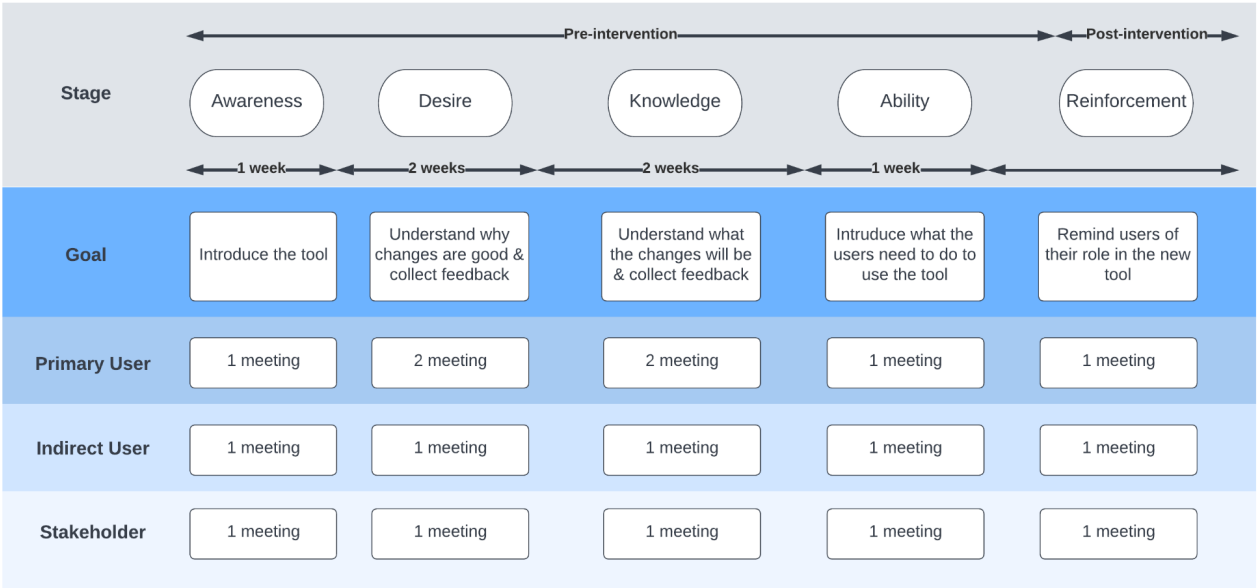

Figure 4. Timeline for ADKAR framework

3) Include any special considerations about things you may need to define about your tool that you will need user input on. Include any prior understanding of what users may care about (e.g. some users may care about the tool being simple and fast to use.)

In order to include the following consideration into our tool design, we will communicate these questions to our primary users during the Desire stage, collect their feedback, and incorporate some good and feasible suggestions into our tool.

Customization options for CT scan

Identify what kind of customization options our primary users may want for the presentation of the CT scan. For instance, whether they want to be able to apply an additional customized window filter on the scan, or whether they would want the tool to always show a slightly large region than the predicted organ bounding box.

User Interface to manually edit organ location

Identify what kind of user interface our primary user prefers to edit the organ location in case the model prediction is not accurate. For instance, whether they prefer simple and easy-to-use UI or more complicated ones with more functionalities.

User Interface to report feedback about model accuracy

What kind of user interface for the report feedback system? Does our primary user prefer choosing from the Likert scale, or if they also want the additional option of entering text comments?

Option to opt of from the tool and use traditional methods

Whether some primary users want to be able to completely opt-out from using this tool and keep the traditional methods to manually evaluate the raw CT scan instead.

Source

Yu, Y., Zhu, Y., Li, S., & Wan, D. (2014). Time series outlier detection based on sliding window prediction. Mathematical problems in Engineering, 2014.

Paper: Navarro, F., Sekuboyina, A., Waldmannstetter, D., Peeken, J. C., Combs, S. E., & Menze, B. H. (2020, September). Deep reinforcement learning for organ localization in CT. In Medical Imaging with Deep Learning (pp. 544-554). PMLR. https://arxiv.org/abs/2005.04974

Paper: Evaluating Reinforcement Learning Agents for Anatomical Landmark Detection https://openreview.net/pdf?id=SyQK4-nsz

Dataset: Multi-Atlas Labeling Beyond the Cranial Vault - Workshop and Challenge https://www.synapse.org/#!Synapse:syn3193805/wiki/89480

Data description:

https://www.synapse.org/#!Synapse:syn3193805/wiki/217789

Colab Link:

https://colab.research.google.com/drive/1e9APrayp6NXjk10kI8ywRtV8kho7tv5F?usp=sharing