Final Report

Predicting hemorrhage related outcomes with CT volumetry for traumatic hemothorax

Benjamin Albert   Chang Yan   Gary Yang

Table of contents:

1. Introduction

Hemothorax (HTX)—blood accumulation in the pleural cavity around the lungs—patients are currently treated by qualitative estimates of blood volume using CT scans. To automate this analysis, deep neural networks are employed to segment hemothoraces from patient CT scans. The network segmentation is converted to an estimated volume, yielding an adjusted R of 0.91 compared with manually segmented volume, done by radiologists. This predicted volume is then used as a predictor for a composite variable: patient requires massive transfusion or dies. Together with clinical data, a random forest classifier achieves an auROC of 0.944, indicating strong predictive capabilities for the composite variable.

2. Background

The current standard for estimating hemothorax volume is a qualitative grading done by radiologists. However, such measurement is subjective and the reliability of measurements relies on radiologists' level of experiences. Even expert radiologists often disagree on the qualitative estimates. In contrast to the qualitative measurement, manual segmentation produces precise, quantifiable blood volume. However, this task is time-impermissible, especially for trauma cases. Because an accurate hemothorax volume estimate can assist physicians in predicting patient outcomes, such as the need for massive transfusion and mortality, there is a need for developing a fast and reliable method to segment hemothorax CT scans and estimate corresponding volume.

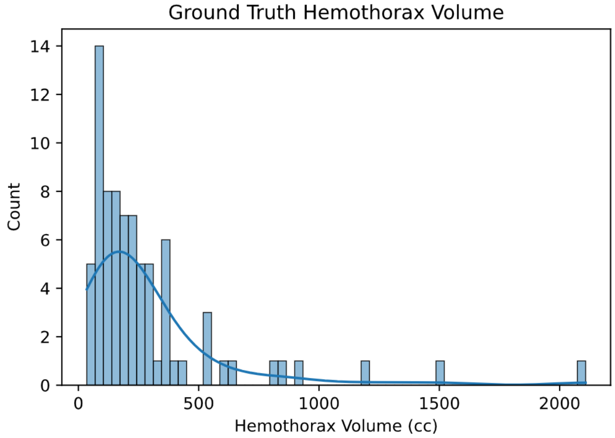
There is no prior attempt for automatic hemothorax volumetry, yet researchers have tried to semi-automate volumetry for pleural effusion, a condition where excess liquid including water and blood accumulates around the lung [4]. Pleural effusion is a comparable condition to hemothorax. However the methods utilized are generally rule-based or atlas based, which cannot sufficiently handle anatomical distortion, heterogeneity of attenuation, and traumatic lung scenes.

We choose to develop a deep-learning model because it has shown to perform well in other segmentation tasks. U-Net [1] and U-Net 3D [2] are two of the most famous deep neural networks in the field of medical image segmentation.

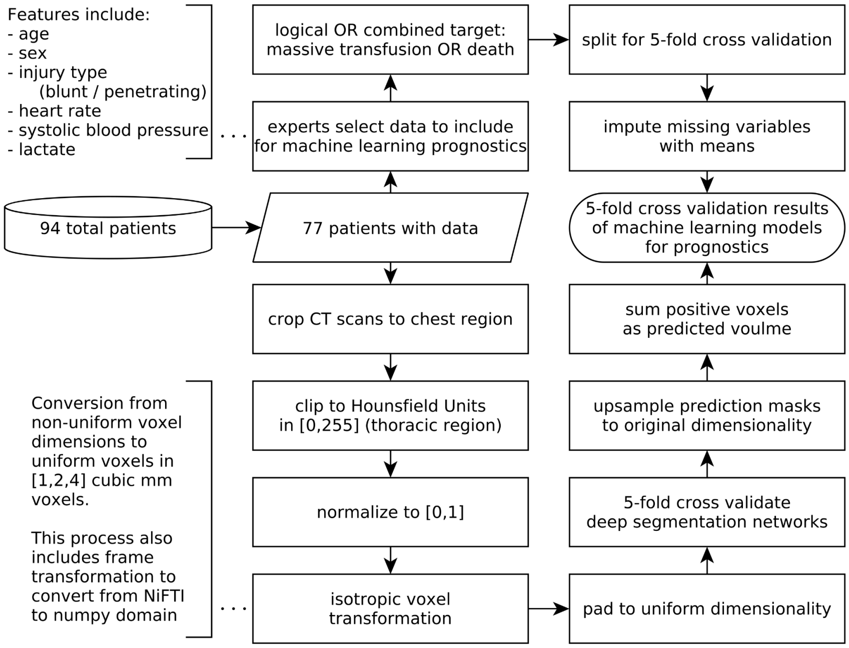
3. Data

The original data comprises 94 patient cases including CT scans, corresponding manual segmentations, and clinical variables. The clinical variables include age, sex, injury type, heart rate, systolic blood pressure, and lactate concentration, which are selected by Dr. Dreizin and his medical student Bryan Nixon; the variables are chosen for clinical relevance as they are available upon admittance to a hospital and have the potential to correlate with prognostic dependent variables. Of the 94 patients, only 78 had valid CT imagery as some had corrupted metadata which prevented the calculation of the ground truth hemothorax volume. After removing these erroneous instances, another case was removed for not having all clinical variables. Therefore, the prognostics dataset used 77 patient cases.

The distribution of the hemothorax volumes is illustrated below:



4. Procedure Overview


Features include:
- age
- sex
- injury type
(blunt / penetrating)
- heart rate
- systolic blood pressure
- lactate
94 total patients
Conversion from non-uniform voxel dimensions to uniform voxels in [1,2,4] cubic mm voxels.
This process also includes frame transformation to convert from NiFTI to numpy domain
logical OR combined target: massive transfusion OR death
experts select data to include for machine learning prognostics
77 patients with data
crop CT scans to chest region
clip to Hounsfield Units in [0,255] (thoracic region)
normalize to [0,1]
isotropic voxel transformation
split for 5-fold cross validation
impute missing variables with means
5-fold cross validation results of machine learning models for prognostics
sum positive voxels as predicted voulme
upsample prediction masks to original dimensionality
5-fold cross validate deep segmentation networks
pad to uniform dimensionality

5. Preprocessing

Dr. Dreizin manually segmented the hemothorax volumes on 94 patient CT scans. Of these 94 CT scans, 79 are suitable for the segmentation task because the other 15 have corrupt metadata in which voxel dimensions are distorted and/or non-sense (e.g. some patients are 12m tall or impossibly proportioned).

After removing these 15 bad data, the data are manually cropped to represent chest CT scans. Most of the original data are full-body CT scans, so approximately 70GB out of the 90GB total are removed through cropping. Scans are cropped on the z-axis below the liver and up to the neck. The axial perspectives are preserved.

The cropped data are then converted from NIfTI format to 3D numpy arrays with isotropic voxels. To transform the NIfTI data, python scripts are used to interpolate data to 1mm cubic voxels, after which they are saved to disk in compressed format, requiring 7-8 GB total for the compressed scans and segmentation masks. Decompressed, the data is approximately 18 GB.

The input is then padded to a fixed size for use in neural networks. To do this, the max shape size of all 79 input volumes is found, after which the data are padded along the borders such that the unpadded volume is in the center of the padded volume. Data are padded with -1024 to approximate the Hounsfield unit for air.

Lastly, the data are normalized in two manners to be used for experimentation: normalization and standardization. Normalization represents a simple linear scaling to the bounds [0,1]. Standardization centers the mean about zero and scales the data to a standard deviation of 1; this method does not bound the data.

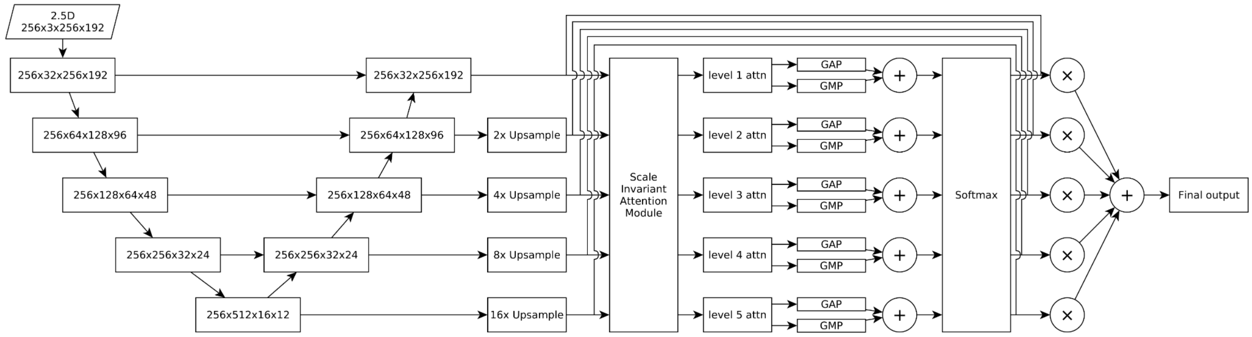
The dataset consists of axial CT scans with 1.5mm voxel resolution from 94 patients. In total, the dataset is approximately 90 GB before preprocessing. Preprocessing involves three primary stages; smoothing, interpolation, and construction of sagittal and coronal perspectives. Smoothing is necessary to fill holes that are erroneously present from noisy manual labelling; it is applied only to the segmentation masks. Trilinear interpolation is used to generate 1mm voxels so that additional sagittal/coronal slices can be generated. This is useful for network training as it smooths the objective functions. However, in total, the preprocessing multiplies the dataset size in memory by 4.5 fold, reaching approximately 400 GB, averaging 4 GB per patient.

Cropping is performed on the remaining instances to make each instance only a chest CT scan so that the axial perspective is preserved while the z-axis is cropped to the neck and below the liver. Additionally, the 1mm cubic voxels are downsampled to 2mm to reduce the memory footprint and padded to 256x192x256. Each instance, including both the scan and the mask, is then 144 MB. Larger networks will require downsampling to 8mm cubic voxels.

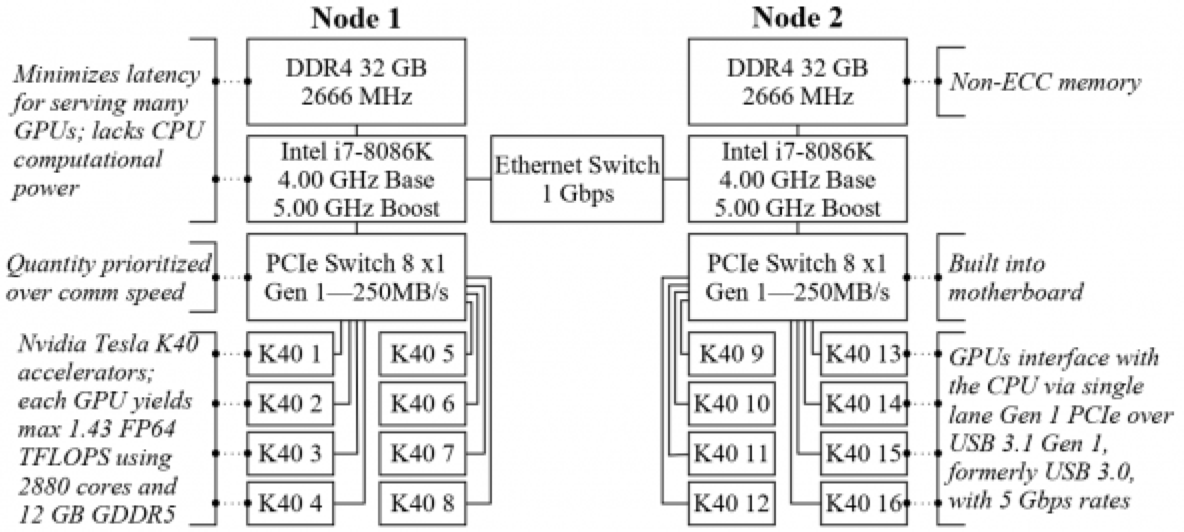
6. Deep Learning

Three deep networks are evaluated: UNet (2.5D) [1], UNet 3D [2], and UNet-FAN. UNet-FAN, the architecture of which is illustrated below, was developed as a combination UNet (2.5D) and PIPO-FAN [3]. PIPO-FAN validation yielded poor performance, so the trained UNet models were used as replacement to the PIPO module to train the FAN scale-invariant attention module post hoc. This transfer learning approach allowed the FAN module to apply attention mechanisms to the multiscale features learned in UNet, slightly improving dice.

It was observed that training deep segmentation networks on left and right lungs individually yielded superior dice scores than from training on the union of the left and right masks. The final predicted volume takes the union of the left and right prediction masks.



All work was conducted on the cluster outlined below:


Node 1
Node 2
DDR4 32 GB 2666 MHz
DDR4 32 GB 2666 MHz
Intel i7-8086K 4.00 GHz Base 5.00 GHz Boost
Ethernet Switch 1 Gbps
Intel i7-8086K 4.00 GHz Base 5.00 GHz Boost
PCIe Switch 8 x1 Gen 1—250MB/s
PCIe Switch 8 x1 Gen 1—250MB/s
K40 1
K40 2
K40 3
K40 4
K40 5
K40 6
K40 7
K40 8
K40 9
K40 10
K40 11
K40 12
K40 13
K40 14
K40 15
K40 16
Minimizes latency for serving many GPUs; lacks CPU computational power
Quantity prioritized over comm speed
Nvidia Tesla K40 accelerators; each GPU yields max 1.43 FP64 TFLOPS using 2880 cores and 12 GB GDDR5
Non-ECC memory
Built into motherboard
GPUs interface with the CPU via single lane Gen 1 PCIe over USB 3.1 Gen 1, formerly USB 3.0, with 5 Gbps rates

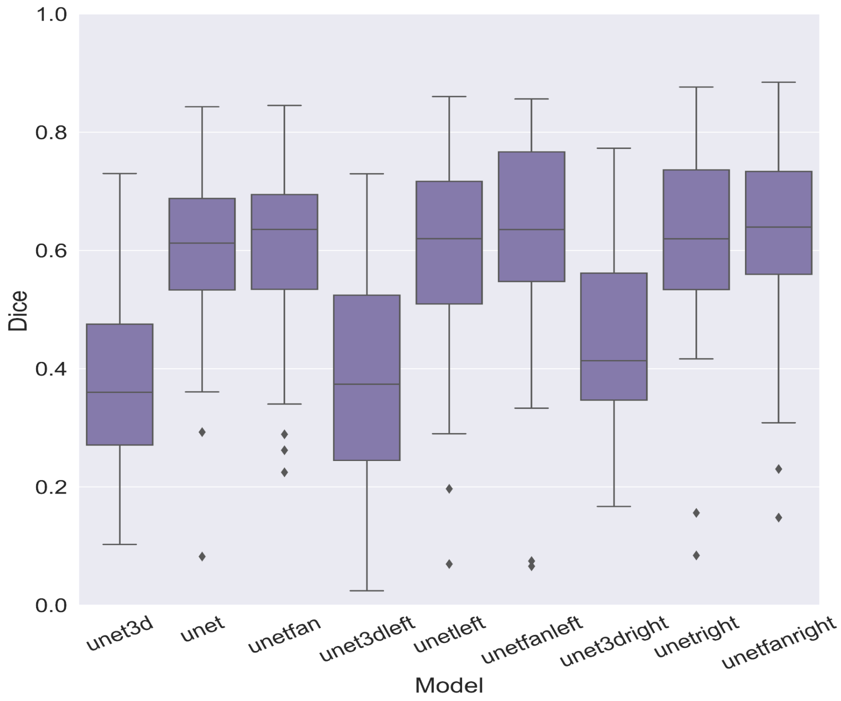


Figure E: Box plot of Dice score of all deep net modes used. UNet-FAN achieves slightly higher dice score than UNet and much better than UNet 3D for all data: left lung, right lung, and the union of the lungs.

7. Machine Learning

Machine learning was applied to predict a composite variable: whether a patient needed a massive transfusion and/or the patient died in the hospital. Univariate analysis was conducted to ascertain the predictive power of the expert volume estimation and the automatically predicted hemothorax volumes for the composite outcome variable. Logistic regression was applied to the independent variable volumes: qual (manual expert volume estimation), U-Net 3D, U-Net (2.5D), and a model called U-Net-FAN, which is a combination of U-Net with PIPO-FAN. The logistic regression results are outlined in the below tables as the average over 5-fold cross validation:

Model	tn	tp	fn	fp	tnr	tpr	fnr	fpr	npv	ppv
qual	59	7	8	3	0.9513	0.4667	0.0487	0.5333	0.8832	0.7333
U-Net 3D	35	12	3	27	0.5615	0.8000	0.4385	0.2000	0.9325	0.3323
U-Net 2.5D	51	9	6	11	0.8218	0.6000	0.1782	0.4000	0.8949	0.5267
U-Net FAN	47	6	9	15	0.7538	0.4000	0.2462	0.6000	0.8608	0.3545

Model	f1n	f1p	mcc	auroc	auprc_0	auprc_1	mae	rmse
qual	0.9149	0.5333	0.4931	0.7609	0.6219	0.6590	0.5350	0.5390
U-Net 3D	0.6817	0.4556	0.3090	0.7692	0.7072	0.4755	0.5354	0.5409
U-Net 2.5D	0.8524	0.5289	0.4155	0.7017	0.7230	0.5319	0.5336	0.5378
U-Net FAN	0.7715	0.3000	0.1746	0.7081	0.7232	0.4900	0.5305	0.5344

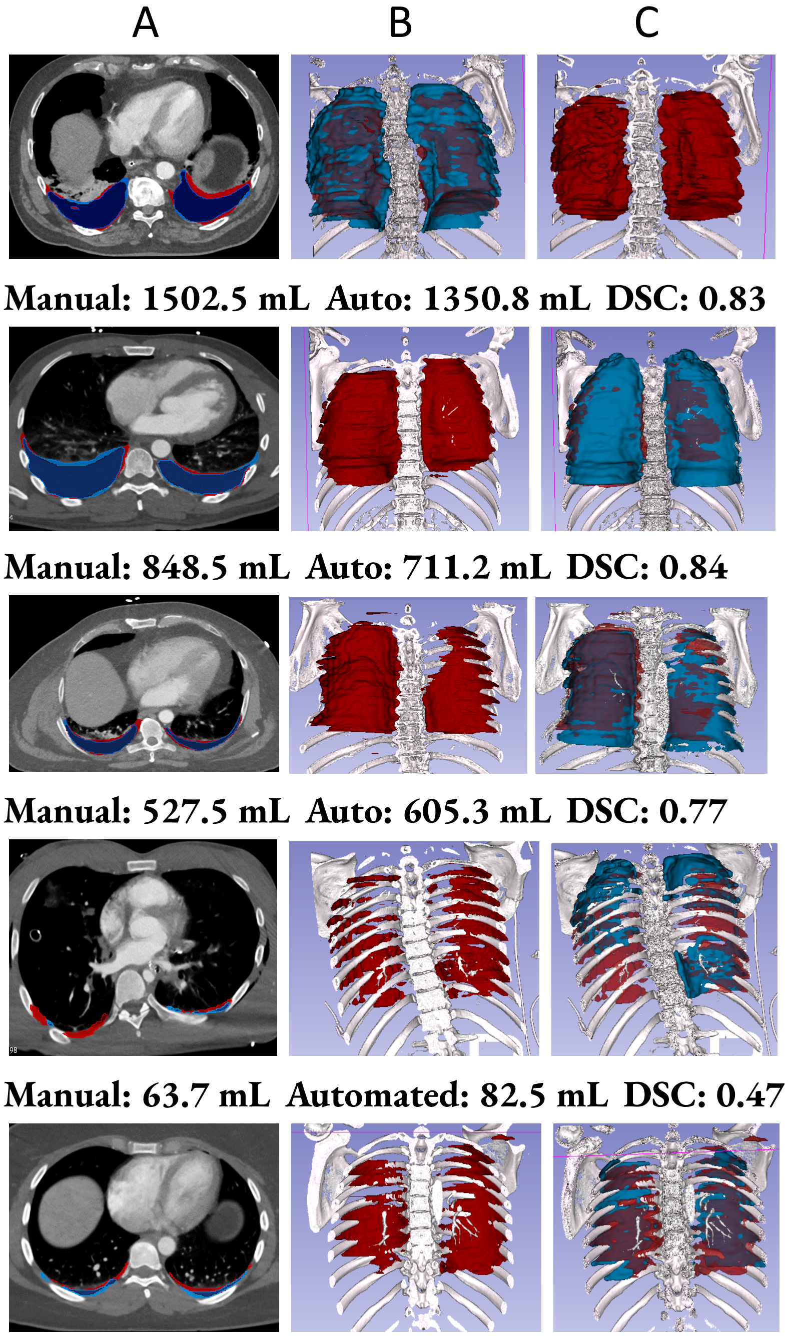
Model	pval	aic	bic	adj_r^2	conf_int (95%)	coeff	thresh_prob	thresh_ml
qual	0.1207	136.5880	139.1851	0.0214	-0.0306 0.3498	0.1596	0.5791	3.0000
U-Net 3D	0.0712	135.7848	138.3819	0.0272	-1.7e-05 6.6e-04	0.0003	0.5697	676.8000
U-Net 2.5D	0.1292	136.1236	138.7207	0.0247	-1.4e-4 2.1e-3	0.0010	0.5816	336.2000
U-Net FAN	0.1416	136.2626	138.8597	0.0237	-2.0e-4 2.2e-3	0.0010	0.5794	318.0000

In addition to univariate analysis, clinical features are used for multivariate predictions. Eight machine learning models [6] are evaluated: logistic regression, bayesian network with global tabu architecture search, discrete naive bayes, gaussian naive bayes, decision table, linear support vector machine, RBF support vector machine, and random forest. The random forest models performed best, demonstrating comparable performance between the manual and automatic features for the composite variable prognostics. The random forest results are detailed in the below tables:

Model	TNR	TPR	FNR	FPR	NPV	PPV	F1N	F1P
qual	0.919	0.733	0.267	0.081	0.934	0.688	0.927	0.71
U-Net 3D	0.919	0.800	0.200	0.081	0.95	0.706	0.934	0.75
U-Net	0.919	0.80	0.200	0.081	0.95	0.706	0.934	0.75
U-Net FAN	0.919	0.800	0.200	0.081	0.95	0.706	0.934	0.75

Model	MCC	AUROC	AUPRC_N	AUPRC_P	MAE	RMSE	Kappa
qual	0.637	0.945	0.986	0.855	0.1731	0.2829	0.6366
U-Net 3D	0.687	0.948	0.986	0.86	0.1735	0.2633	0.6847
U-Net	0.687	0.932	0.979	0.85	0.1836	0.2785	0.6847
U-Net FAN	0.687	0.944	0.987	0.768	0.1835	0.2831	0.6847

8. Result Visualization



Manual: 1502.5 mL Auto: 1350.8 mL DSC: 0.83

Manual: 848.5 mL Auto: 711.2 mL DSC: 0.84

Manual: 527.5 mL Auto: 605.3 mL DSC: 0.77

Manual: 63.7 mL Automated: 82.5 mL DSC: 0.47

Manual: 80.6 mL Auto: 80.2 mL DSC: 0.46

(A) Overlap (purple) of manual and automated (FAN UNET) hemothorax labels on axial images. (B) 3D rendering of automated label. (C) 3D rendering of overlap (purple) between automated (red) and manual (blue) labels.

9. Result and Analysis

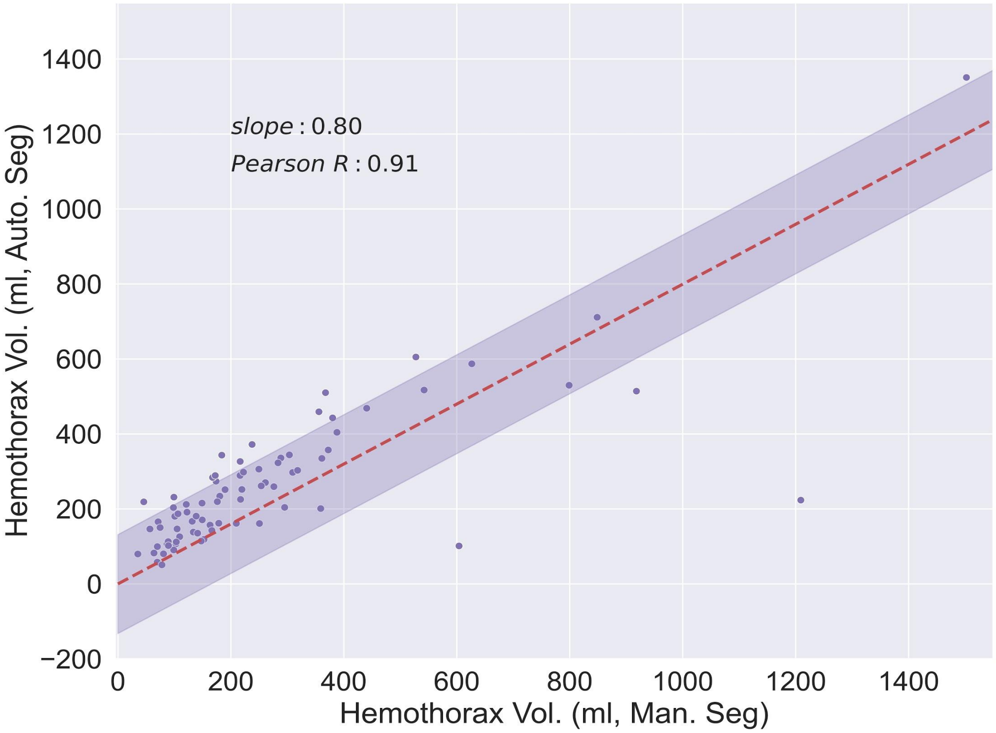


Figure A: Dot matrix plot with best-fit line and 95% CI shows correlation between automated volume (vol.) and manual hemoperitoneum volume. The prediction from human experts and our deep learning is consistent.

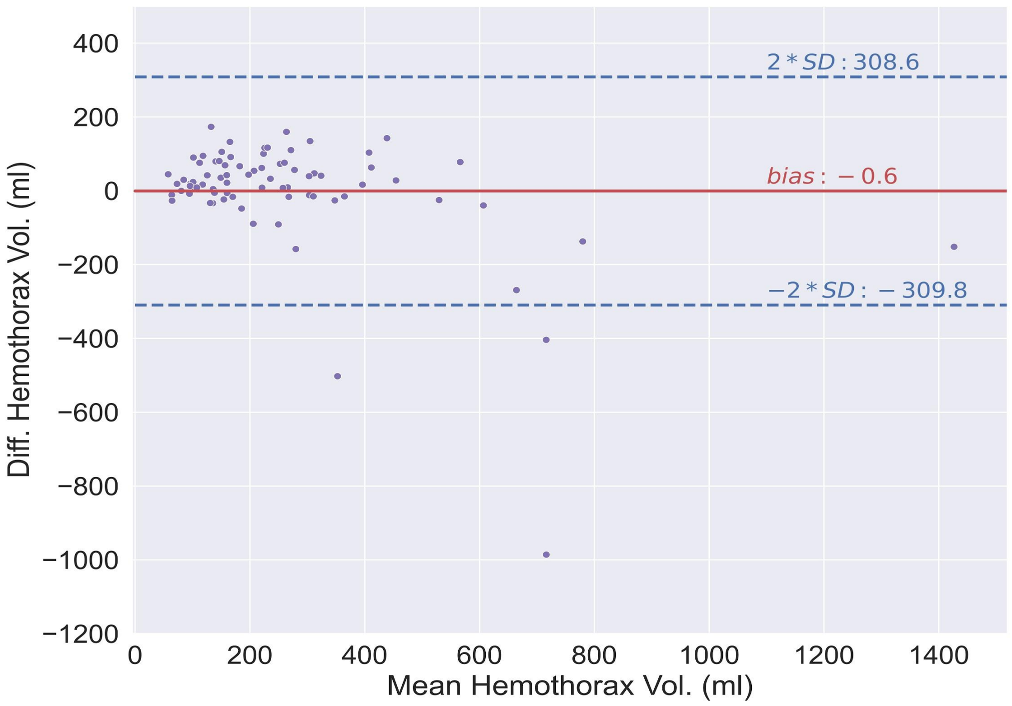


Figure B: Bland-Altman plot shows 95% limits of agreement and measurement bias. On average, there is a 0.6-mL underestimation by the deep learning algorithm. The bias is relatively small and standard deviation is 155.6 mL.

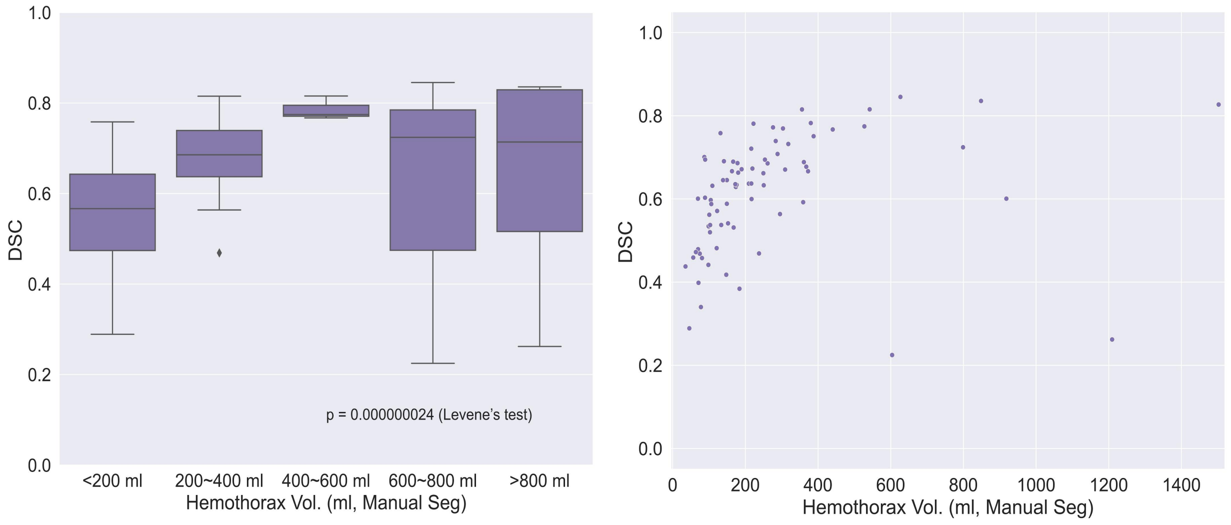


Figure C: Distribution of Dice similarity coefficients (DSCs). The box plot in C2 shows DSC improves/variance decreases with increasing vols at volume range 0-600 ml, (Levene's test, $p < 0.00001$) explaining low DSCs in rows 4 and 5 (image left). In volume range >600ml, we have only 7 instances and some of them are outliers, so the deep network does not learn this range well so behavior in this range is not clear.

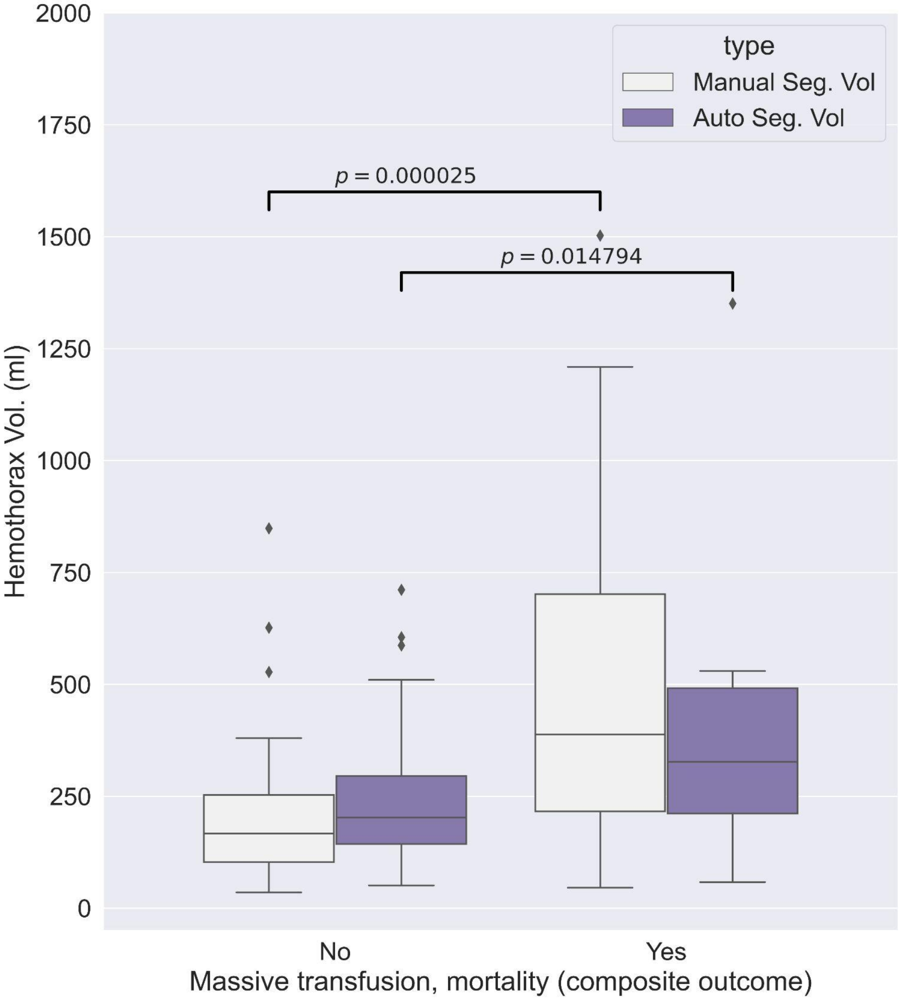


Figure D: Clustered box and whisker plots show prediction of a composite outcome for the need for massive transfusion and in hospital mortality. Manual and HTXvol-auto vols both have significant association with composite outcome (MT + IHM), with p = 0.0003 and 0.015 respectively.

In general, the deep network predicted volume and manual segmented volume are highly associated with adjusted R=0.91 and the bias is very low at -0.6 mL. The Dice similarity coefficient improves and its variance decreases as volume increases. The small hemothoraces with lesser dice scores are clinically insignificant compared to the larger accumulations of blood. Therefore, it is important that the performance of the automated volume estimates is best for larger hemothoraces.

Both manual and predicted volume have significant association with the requirement for mass transfusion and in-hospital mortality.

We are able to predict the composite outcome of MT+IHM using automated prediction volume and 6 patient metadata (Age, Sex, HR, BP, lactate, injury-type: blunt / penetrating) with random forest model and reach an auROC of 0.9440. This is at least as good as using expert information from 2 radiologists.

The results suggest that the automated methods can replace expert analysis with comparable performance, thereby reducing costs, labor, and improving availability to accurate prognostics.

10. Management

a) Who did what

Project plan and data preparation: All of us
Preprocessing and file IO: Chang Yan, Gary Yang
Pipeline and APIs: Benjamin Albert, Chang Yan
Normalization and Padding: Chang Yan, Benjamin Albert
Training: UNets (3D/2.5D/2D): Gary Yang, Chang Yan
Training: Attention Nets (3D/2.5D): Benjamin Albert
Training: SCMs: Chang Yan
Training: Multiscale Net (2.5D): Benjamin Albert
Training: Unet-FAN (Final model): Gary Yang
Evaluation API: Chang Yan, Gary Yang
Result collection and analysis: All of us
Result and data visualization: Chang Yan
Final poster, presentation and report: All of us

b) Meetings

As we had planned, the student team meets with each other almost everyday through WeChat group or Zoom meetings. Progress is relayed to the mentors every couple of weeks via text, email and zoom meetings. We had regular calls and texts with Dr. Dreizin with a frequency higher than twice a week, and Dr. Dreizin discussed our result with Dr. Unberath regularly. We had zoom meetings with both mentors at each milestone time.

c) What we accomplished vs planned

The “Minimum” part has been fully done as expected, including literature research, preprocessing, APIs and model benchmarking. Preprocessing was a little over time because of the severe data corruption we faced, and we had to manually check and correct all data.

“Expected” part has been changed according to the benchmark result. The original plan of doing ensemble has been replaced by trying more attention and multiscale modules, and trying to combine them with Unet and transfer learning to get a higher result. The structural causal models have been tested in various ways, and they turned out to not be suitable for solving our model mostly due to the hardship of discovering strong causal relationships from our data.

"Maximum" part has also been changed due to the lack of time and the new clinical data we received. We decide to cancel the GUI part which is not essential to our project, and replace it with implementing a model to predict the composite outcome of patient need for massive transfusion and their in-hospital mortality rate using the predicted volume and additional clinical data.

Details will be discussed in the "Deliverables" part later.

d) Future steps

1. Computing loss for each decoding layer
2. Adding data augmentation to counter the lack of large HTX cases
3. Real-world clinical application

e) What we learned

1. Time estimates were too short, particularly for the maximal deliverables
2. Agile development was more effective than waterfall methods
3. Concentrating developer time on related tasks was most effective
4. Making scripts/executables flexible with respect to the environment, such as directory structures and command line arguments, was important

11. Deliverables

Nearly half of our expected/maximum plan and deliverables has been adjusted multiple times according to the milestone result and suggestion from our mentors. This is mostly due to two reasons: First, our maximum plan on GUI and other visualization was too ambitious and there is no way they can be done in one month, so they are cancelled and we switched to do the composite outcome prediction as suggested by our mentors. Second, there are actually many choices to improve the benchmark models to get a higher result, and only the benchmark analysis itself can tell us which one is best to try. Thus, we switched from doing an ensemble to a combined multiscale network with transfer learning.

	Activities	**Result**	**Deliverables**	**Due**	**Status**
Minimum	Literature survey for model selection	Draft a list of open-source models with code or architecture description	Plan to test Unet, attention nets and SCMs in 2D/2.5D/3D	2/22	**Done**
	Preprocess CT scans (interpolate, make 3d slices)	Interpolate CT scans and convert data to PyTorch tensor type	Data stored on server	3/1	**Done**
	Complete pipeline and I/O APIs for the project	Build a network framework consists of Python classes	Code with documentation in private repo	3/1	**Done**

	Benchmark open-source models	Benchmark existing open-source models measured with Dice/Jaccard	Result reported in excel sheet	3/21	**Done**
Expected	Research on and implement several Deep Structural Causal Models	Several implemented SCM	2 SCM implementations, none outperforms other models	3/26	**Done**
	Research on and implement several multiscale Models	Implemented PIPO-FAN deep network	Code with documentation in private repo, result in excel sheet	4/25	**Added** **Done**
	Implement a combined model that outperforms others as final model	Implemented Unet-FAN combined deep network with full analysis	Code with documentation in private repo, result in excel sheet. Also has a visualization report.	4/25	**Added** **Done**
	Design and implement an ensemble algorithm	A documented program that estimates blood volume	Cancelled due to benchmark analysis and suggestions from professors	4/4	**Cancelled**
	Improve the ensemble algorithm	A documented program outperforms the benchmark	Cancelled due to benchmark analysis and suggestions from professor	4/11	**Cancelled**
Maximum	Predict the requirement for mass transfusion and in-hospital mortality rate using predicted volume and clinical data	A trained model that predicts the outcome of each patient using their predicted volume and 6 clinical datas	Code with documentation in private repo, result in excel sheet.	4/30	**Added** **Done**
	Incorporate certainty level into our algorithm	A documented program visualizes confidence	Cancelled due to time	4/30	**Cancelled**
	Implement a GUI-program for visualization	A documented program incorporates the framework	Cancelled due to time	4/30	**Cancelled**

12. References

[1] Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation. InInternational Conference on Medical image computing and computer-assisted intervention 2015 Oct 5 (pp. 234-241). Springer, Cham.

[2] Çiçek Ö, Abdulkadir A, Lienkamp SS, Brox T, Ronneberger O. 3D U-Net: learning dense volumetric segmentation from sparse annotation. InInternational conference on medical image computing and computer-assisted intervention 2016 Oct 17 (pp. 424-432). Springer, Cham.

[3] Fang X, Yan P. Multi-organ segmentation over partially labeled datasets with multi-scale feature abstraction. IEEE Transactions on Medical Imaging. 2020 Jun 9;39(11):3619-29.

[4] Sangster, G. P., González-Beicos, A., Carbo, A. I., Heldmann, M. G., Ibrahim, H., Carrascosa, P., Nazar, M., & D'Agostino, H. B. (2007). Blunt traumatic injuries of the lung parenchyma, pleura, thoracic wall, and intrathoracic airways: multidetector computer tomography imaging findings. Emergency radiology, 14(5), 297–310.

[5] Dreizin, D., Zhou, Y., Fu, S., Wang, Y., Li, G., Champ, K., Siegel, E., Wang, Z., Chen, T., & Yuille, A. L. (2020). A Multiscale Deep Learning Method for Quantitative Visualization of Traumatic Hemoperitoneum at CT: Assessment of Feasibility and Comparison with Subjective Categorical Estimation. Radiology. Artificial intelligence, 2(6), e190220.

[6] Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH. The WEKA data mining software: an update. ACM SIGKDD explorations newsletter. 2009 Nov 16;11(1):10-8.