Tissue Fate Prediction in Acute Stroke based on MRI

Benjamin Brandt
Computer Science Department
University of California, Los Angeles
Los Angeles, USA
benjibrandt@ucla.edu

Roy Lin
Computer Science Department
University of California, Los Angeles
Los Angeles, USA
rlin2k1@gmail.com

David Julius Macaraeg
Computer Science Department
University of California, Los Angeles
Los Angeles, USA
dmacaraeg@g.ucla.edu

Index Terms—fluid attenuated inversion recovery, machine learning, magnetic resonance imaging, stroke, tissue fate prediction

I. Introduction

STROKE kills about 140,000 Americans every year, meaning 1 out of every 20 deaths in the United States are stroke-related [1]. Put in a more startling way: someone in the United States has a stroke every 40 seconds, and, every 4 minutes, someone dies of stroke [1]. It is a serious, widespread issue, touching most people's lives. Prevention research and specific preemptive drug treatments are making great strides in lowering the risk of strokes, but the reality is that they still happen. Therefore, treatment of ischemic stroke patients is an absolute priority - particularly with a focus on maximizing the recovery of affected tissue. Integral to tissue recovery is the usage of tissue outcome prediction - the process of determining which stroke-affected areas may survive - to inform clinical decision-making. As such, given the common-place nature of strokes, and the importance of predicting their survivability, deducing better methods of prediction is absolutely imperative.

Conventional methods of obtaining images for the prediction of tissue outcome after the treatment of ischemic stroke patients include Computed Tomography (CT) Scan and Magnetic Resonance Imaging (MRI) Scan [5]. Both generally use techniques consisting of Cerebral Angiography or Source Perfusion Imaging [6]. Cerebral Angiography uses contrast material to produce detailed pictures of major blood vessels in the brain including blood clots or narrowing arteries [7]. The scans are timed perfectly with the arrival of the contrast into the brain [2]. Perfusion Imaging, on the other hand, use nonradioactive substances through the blood vessels and takes scans over time to see what areas of the brain got the most blood - giving information about blood blow throughout the brain [3]. Grey areas indicate occluded or clotted blood flow that comes with negative side effects and symptoms.

Three days after surgery to open up the blood vessel physically, another MRI Scan will be performed to see possible post-op damage. Fluid Attenuated Inversion Recovery (FLAIR) images are produced from this MRI Scan to annotate lesions post-surgery [5]. For this study, we will use Source Perfusion Weighted MRI scans obtained after the admission of the stroke patient and Machine Learning to predict possible Lesion growth after surgery. The main advantage of Machine Learning approaches to such images is that we can take advantage of a large number of previous surgeries and estimate future surgeries based on largely non-linear functions. These functions are obtained by methods including Linear Regression, Decision Trees, K-Nearest Neighbors, Logistic Regression, Support Vector Machines (SVMs), and Mixture Models.

Previously, we would rely on doctors analyzing contrast-intensity time curves, which have a lot of noise and motion artifacts [8]. Manual analysis is time intensive and can be unreliable. In current practice, programs use the spatial differences between diffusion and perfusion-weighted MRI to differentiate irreversible blockage of blood supply from salvageable tissue. Early models of tissue outcome based on computer vision and pattern recognition techniques have been trained on a voxel-by-voxel basis using perfusion imaging. However, more recently, relative to single-voxel-based methods, models of tissue prediction that use data in the surrounding area of the target have been shown an improved accuracy in relation to the predictive model. [10]

In Machine Learning, parameters are continually adjusted to achieve better performance with new training data. Past research studies on Hemorrhage Severity in the Brain have shown high accuracy using Machine Learning Models [9]. We will use Lesion labels and PWI - MRI Scans from past surgeries to predict Lesion labels for new surgeries.

Ultimately, the aim of this paper is singular: to examine the various machine learning methods of tissue outcome prediction, rank them, and determine which are best suited. By best suited, we mean the methods which have the highest rates of successful prediction, the most amount of the time; put shortly: the most consistently accurate. To achieve this goal, we utilize multiple machine learning algorithms available in the sci-kit learn package. First, we train them on sets of images such that they can observe before and after shots of brain tissue. Then, we allow them to operate on a broader set of images, this time predicting various tissue regions' survival via a binary survive or did not survive. We then compare the algorithms' predictions with the actual results of survival, and determine the accuracy. Clearly, more accurate is better, and so, in this way, we can deduce which algorithm is best suited to this problem domain. By creating such a ranking, we hope to create a reference for clinicians to use to better inform their decisions regarding tissue outcomes. Knowing the strengths and weaknesses of each machine-aided analysis, and their relative accuracy, will give decision-makers a greater context on which to base their understanding, and their ultimate choices. With clear-eyed, open data regarding various methods' effectiveness, clinicians will be able to more scientifically formulate plans of action, ultimately improving patient care and outcomes.

II. Steps Taken

II-1 Overview

First, we detail a broad overview of our process. To start, we were given a series of MRI scans for eighteen patients who had been treated for ischemic stroke at UCLA's Ronald Reagan Medical Center. Each patient had two sets of MRI images: perfusion, and FLAIR, represented in the international standard DICOM image format [11], which is essentially a standard image appended with a load of metadata, such as patient name, brain slice location (slice meaning vertical location, relative to the patient's eyes,

Manuscript revised June 7, 2019. Current version June 7, 2019.

moving up or down), and image capture time. Perfusions were taken immediately after the patient suffered from a stroke, and are a series of full-brain images, taken over time. They are taken over time to reveal how a contrasting agent – injected just before the imaging process begins – flows through the brain's major blood vessels, indicating which areas are flowing freely, and which areas are occluded or clotted. This allows each voxel (a 3D pixel, composed of X and Y coordinates across a slice, with Z indicating slice location) [12] within the brain scan to be viewed by its intensity over time – meaning, how bright or dark that voxel gets as time passes. This voxel-intensity relationship is what we used to train the machine learning model, to predict post-operation lesion formation.

On the other hand, the FLAIR images are one single full-brain scan - taken three days after surgery to open up the occluded vessels. These indicate where lesions occurred post-surgery, giving us hard answers (labels in machine learning lingo) about the outcomes of individual voxels. In this way, we could give the machine learning model a series of voxels paired with their intensity values over time, as well as the answer of whether the voxel would develop into a lesion or not, and train the model to associate certain intensity curves with certain outcomes.

As an aside: please note that any references to specific script names refer to the code on our GitHub repository [13]. We encourage readers to look at the documentation in this repository for even more detailed instructions on how to use our materials, and produce results.

II-2 Slice Counting and Alignment

Now, we examine our process in more detail. First, we created a python script, `FLAIR_perfusion_counter.py`, which took in the DICOM files for each patient and counted the number of unique slices in both the perfusion image set and the FLAIR image set. We did this so that we could obtain accurate results by comparing post and pre-operation data from slices at the same level. To go about this process, we conducted initial filtering: if the slice numbers were the same, we kept the patient's data full-stop, without further intervention. If they were not the same, we did one of two things: if there were more slices in the perfusion set than the FLAIR set, we removed the patient's data from our consideration; if there were more FLAIR slices than perfusions, we took a manual approach to align the slices. We threw away patient sets with a greater number of perfusion slices because speed was of the essence, we had enough data without these few patients (two, in total) and we did not have a good method to properly align the thousand-or-so slices belonging to a perfusion set, were they the ones with extraneous data.

To manually align sets that contained more FLAIR slices than perfusion slices, we combed through the FLAIR images, and a matching set of images representing a full brain from the perfusion images, using Osirix Lite [14]. We then found the one slice in both the FLAIR and the perfusion sets where the ventricle sizes matched the best – we took this as the matching baseline. We then iterated along both the FLAIR and perfusions in unison, one at a time, finding the point where the perfusions ended, and the FLAIRs had extraneous images. We then deleted the trailing, extraneous FLAIR slices, so that the number of slices for perfusion and FLAIR matched. In this way, the slices were aligned exactly, making each FLAIR slice have a direct correspondence in the perfusion set.

II-3 Lesion Labeling and Co-registration

Once each patient's set of slices had been aligned, we then manually labeled the lesions present on the FLAIR images for each patient using HOROS [15]. This meant that we explicitly drew out the areas where lesions developed, demarcating them with a bright color. With the labels created, we then created a new DICOM representation of the FLAIRs, containing masks – the areas we had labeled were given voxel intensity values greater than zero, and every other voxel in the image set was given an intensity value of zero. For ane example, please see Figure 1.

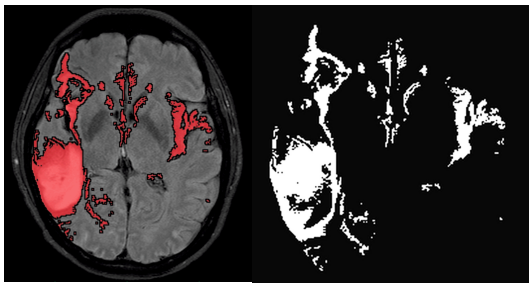

Figure 1: a hand-labeled FLAIR image (left) and an associated mask (right).

Next, because the FLAIRs were taken in a separate session from the perfusions, the coordinate axes for the images did not exactly line up – meaning, one voxel in a perfusion did not directly relate to a voxel with the exact same coordinates in a FLAIR. To correct this issue, we used a program called SimpleElastix [16] to co-register the two image sets – meaning, we transformed the FLAIR images to use the same coordinate systems as the perfusions, allowing a direct comparison of voxels across perfusion and FLAIR images. To see an example of this, please look at Figure 2 below.

Note that the slice counting and alignment processes were done on mid-2018 MacBook Pros, running quad-core, Intel Core i5 processors at 2.30 GHz, each with 16 gigabytes of DDR3 RAM. The co-registration, which was much more computationally intensive, was run on a Google Cloud virtual machine instance, utilizing 8 quad-core, Intel Zeon processors at 2.30 GHz, with 52 gigabytes of DDR3 RAM [17].

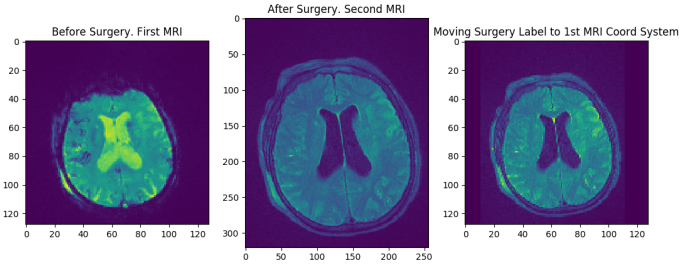

Figure 2 : Co-registration of an after-surgery FLAIR to the before-surgery perfusion.

II-4 Training Data Generation

With the FLAIR images labeled, masked and co-registered to the perfusions, we then utilized our Python script `generate_csvs.py` to generate training data for the machine learning model. This script read in the entirety of the perfusion image set per patient and obtained the set of intensity values for each individual voxel over time. Please see Figure 3, which displays examples of such intensity curves.

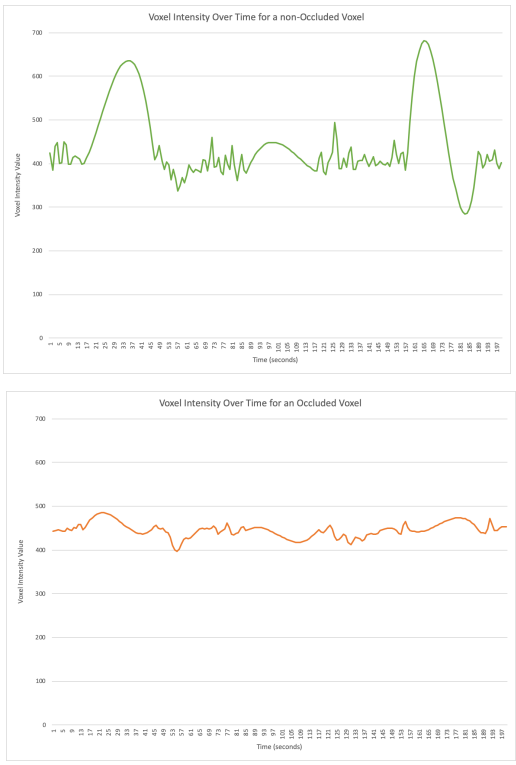

Figure 3: intensity curves for selected voxels. An occluded or clotted intensity curve (bottom) will look flat, as the contrasting agent cannot flow through. An unblocked voxel (top) will have an intensity curve with at least one large bump in it – indicating free flow of the contrast agent.

Once a curve had been constructed for each voxel, the script used cubic interpolation (filling in gaps between data points, to form a continuous function) to generate functions representing those curves, and sampled those interpolations across evenly-spaced intervals, to ensure comparable, same-length levels of data for each voxel across every patient. It then associated each voxel, and its paired sampled intensity values, with a binary outcome of either: lesion, or no lesion. It did this by a simple one-to-one comparison of the `X,Y,Z` coordinates of the perfusion voxel with the `X,Y,Z` coordinates in the FLAIR mask, utilizing the mask value set earlier to determine the presence of a lesion. With this done, an even sampling of 250 voxels which did not get lesions, and 250 which did was created, and written out to a comma-separated value file, with the form: `Lesion,IntensityArray`. As an important aside, please consider the fact that the intensity values were not at a single, consistent baseline across patients. Before feeding them to the machine learning model, we had to normalize the intensity values by rescaling each one relative to the rest, obtaining a value between 0 and 1 [18].

We feel it is important to note that, because this data processing step was so computationally expensive, we performed it on another Google Cloud virtual machine instance, this one utilizing 24, 12-core Intel Xeon processors at 2.30 GHz, with 90 gigabytes of DDR3 RAM.

II-5 Training the Models

This CSV was used to both train and test the various machine learning models: the intensity values and associated labels were segregated into 80% training, and 20% testing via a K-folds process, where we set K to 50 [19]. The K-fold process is a resampling technique to divide the dataset into K folds (groups). We used 80% of the folds to train, and 20% to test – meaning, we assigned 20% of the groups as data to test the models' accuracy, and 80% to train them. We repeated this process 50 times, alternating the composition of the training and testing groups to get an averaged result. Table 1 indicates the various models used, their averaged F1 accuracy on the training data described in the K-folds process, and their averaged F1 accuracy on the testing data described.

Note that F1 score is the weighted average of the precision and recall values. Precision is the number of true positives (the number of values that were guessed to be lesions, that were lesions) divided by the combined total of true positives and false positives (false positives being the number of values guessed to be lesions, but weren't lesions). Recall is the number of true positives divided by the combined total of true positives and false negatives (values guessed to not be lesions, which actually were). We used the F1 Testing score as the decider, as opposed to a pure accuracy metric, because accuracy by itself is artificially inflated, since the majority of a brain will not have developed lesions, and so the model will, by default, be highly "accurate" based on the fact that most of the voxels will be successfully identified as not developing lesions [20].

Training data is the data used to fit our model, while the testing data is used to evaluate our model on "unseen" data (the data the model has not yet encountered). We calculated the F1 score on both training and testing data to see if there was any overfitting to the training data – meaning, it fit so well to data it had already seen, that it did not adapt well when faced with unseen data. However, the F1 scores are close enough to each other that no overfitting was detected.

As is obvious from Table 1, we saw that the multilayer perceptron activation (MLP) logistic classifier, and the gradient boosting classifier, had the highest average F1 Testing scores over the 50 fold run, and thus we utilized those two models for further testing.

III. Prediction Efforts

With the two models selected, we trained them with a 100/0 split – meaning, all our sample data was used to train the model, none was split into an "unseen" portion. Once the models were trained, we had them attempt to predict the lesion outcomes of two patients using CSV data in the form of `X,Y,Z,IntensityArray`, where `X,Y,Z` represented every single voxel across a patient's MRI scans, along with their associated intensity curve. The model output a CSV of the

form `X,Y,Z,PredictedOutcome`, where `PredictedOutcome` was a 0 or 1 value, indicating: no lesion, or lesion, respectively.

Table 1: Machine Learning Models' Initial F1 Scores

Model Name	F1 Score -- Seen/Training Data (%)	F1 Score -- Unseen/Testing Data (%)
Bagging Classifier	100	72
Gradient Boosting Classifier	98	76
Logistic Regression Classifier	55	55
MLP Classifier with Identity Activation	51	51
MLP Classifier with Logistic Activation	78	76
MLP Classifier with TanH Activation	78	74
MLP Classifier with Relu Activation	66	66
Nearest Centroid Classifier	65	66
5-Neighbors Classifier	100	72
Decision Tree Classifier	100	67
Random Forest Classifier	100	73
Extra Trees Classifier Classifier	100	73
Stochastic Gradient Descent Classifier with Hinge Loss	55	55
Stochastic Gradient Descent Classifier with Log Loss	53	53
Stochastic Gradient Descent Classifier with Perceptron Loss	66	67
Stochastic Gradient Descent Classifier with Modified Huber Loss	45	46
Stochastic Gradient Descent Classifier with Squared Hinge Loss	62	63
Support Vector Classifier with RBF Kernel	44	44
Support Vector Classifier with Sigmoid Kernel	58	57

We recorded an F1 score for each model for this process, giving us: 78.65% for MLP activation logistic regression and 94.94% for the gradient boosting model.

Then, we used our Python script, `label_highlighting.py`, which reads in a model's outputted CSV, and generates a color mask over the original perfusion image, coloring where the model has predicted lesions. An example of this color mapping, along with the original perfusion, and the actual FLAIR outcome, are displayed below.

IV. Results

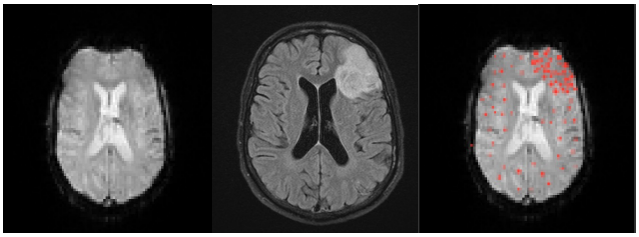

Figure 4: accurate painting. Perfusion (input data, left), FLAIR (labeled data, middle), Predicted FLAIR (target output, right).

Clearly, based on Table 1, and the F1 score of 78.65% for the 100/0 split on the MLP activation logistic regression, and 94.94% for the 100/0 split on gradient boosting classifier, our approach resulted in what we would term a decent level of prediction accuracy. We were able to detect lesion development in various parts of the brain, though the model would sometimes mis-predict sections, both large and small. For instance, the example we showed above was of a slice that our model predicted fairly well. However, other slices obtained less-than-stellar results, as detailed in the image set below.

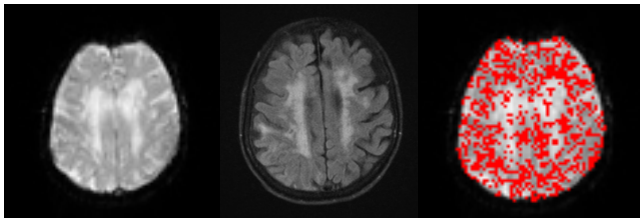

Figure 5: inaccurate painting. Perfusion (input data, left), FLAIR (labeled data, middle), Predicted FLAIR (target output, right).

V. Evaluation

Obviously, these results are not perfect: some predictions are spot on, and others leave a little to be desired. This was clearly indicated by the F1 scores of the models we chose, ranging between 50 and 80, generally. A large bright spot is that the two initially-best-performing models had whole-testing F1 scores of 78.65%, and 94.94%, as previously mentioned. All in all, with only 16 patients used as training data, these are pretty promising results, and we believe that they would only continue to improve with more data, further training, and refinement. After all, the more data is fed into the model – good data, in particular – the better the model will be.

In terms of the overall goal of this paper – which was to compare the accuracy of various machine learning models

in predicting lesion development post-stroke – we think we have done quite well. We now have a general hierarchy of machine learning model accuracy, which can be seen in the table below. Note that this data was obtained via a classification of the initial 80/20 testing scores – with the top-two highest scorers being placed in Tier I, all those scoring in the 70th percentile being placed in Tier II, all those in the 60th percentile placed in Tier III, those in the 50th percentile placed in Tier IV, and anything with an F1 score less than 40% being placed in Tier V.

Table 2: Hierarchy of Machine Learning Models

Hierarchy Classification	Model Names
Tier I: highly accurate initial results	1. Gradient Boosting Classifier 2. MLP Classifier with Logistic Activation
Tier II: accurate initial results	1. 5-Neighbors Classifier 2. Bagging Classifier 3. Extra Trees Classifier Classifier 4. Random Forest Classifier 5. MLP Classifier with TanH Activation
Tier III: moderately accurate initial results	1. Stochastic Gradient Descent Classifier with Squared Hinge Loss 2. MLP Classifier with Relu Activation 3. Nearest Centroid Classifier 4. Decision Tree Classifier 5. Stochastic Gradient Descent Classifier with Perceptron Loss
Tier IV: semi-inaccurate initial results	1. MLP Classifier with Identity Activation 2. Stochastic Gradient Descent Classifier with Log Loss 3. Logistic Regression Classifier 4. Stochastic Gradient Descent Classifier with Hinge Loss 5. Support Vector Classifier with Sigmoid Kernel
Tier V: inaccurate initial results	1. Support Vector Classifier with RBF Kernel 2. Stochastic Gradient Descent Classifier with Modified Huber Loss

With this data, we can now determine that some machine learning models are absolutely better-suited to the task of post-stroke lesion prediction, and we can focus more time and resources into training those models to be hyper-accurate. Specifically, we want to focus on all the models contained within Tiers I and II, as we believe those showed the most promise.

VI. Discussion

Overall, we believe our results provide a positive step forward. They are a good baseline of model classifications, and will provide for a more concerted effort into utilizing the more-successful algorithms going forward. Currently, these results are somewhat flimsy in the sense that there is not an overly large amount of data backing them up, but we anticipate future work will hone in on Tier I and II models, and provide them with an enormous amount of data to increase their predictive power.

One problem with this approach, however, is the data required: as best as we can tell, obtaining truly real-world-ready results will require hundreds, if not thousands, of patients. Computing the data for all of these patients, and incorporating said data into the various machine learning models, will take quite a bit of computing power, and time. However, we think the investment is well worth it: with model selection pared down to a chosen few that work well with this task, we are one step closer to achieving the goal of highly accurate predictions which can streamline clinical decision making. Particularly, we imagine a future where a single clinician can look at prediction, and trust the data enough to use it as a baseline to determine whether or not treatment is necessary – weighing the cost-benefit analysis of surgery versus potential complications. Ideally, the model will be so accurate that hundreds – if not thousands – of patients can be hand-processed by a single trained professional in a given day, essentially offering a final, human-eye "signoff" on the model's predictions. For now, this is only imaginings, but it is tantalizingly close to being within our grasp.

[1] U.S. Centers for Disease Control and Prevention, "Stroke Statistics," U.S. Department of Health and Human Services. September 2017.

[2] Jamary Oliveira Filho, PhD, "Neuroimaging of acute ischemic stroke", UpToDate, Inc., March 2018.

[3] Johns Hopkins Medicine, "Brain Perfusion Scan," *Treatments, Tests and Therapies*. The Johns Hopkins University, The Johns Hopkins Hospital, and Johns Hopkins Health System.

[4] Mayo Clinic, "Stroke," *Diseases and Conditions*. Mayo Foundation for Medical Education and Research.

[5] Segawa F., Kishibayashi J., Kamada K., Sunohara N., Kinoshita M, "FLAIR images of brain diseases. Fourth Dpeartment of Internal Medicine, Toho University, Tokyo, Japan: U.S. National Library of Medicine. June 1994.

[6] Akmal Sabarudin, Cantiriga Subramaniam, Zhonghua Sun, "Cerebal CT angiography and CT perfusion in acute stroke detection: a systematic review of diagnostic value," Diagnostic Imaging and Radiotherapy Programme, School of Diagnostic and Applied Health Sciences, Faculty of Health Sciences, Universiti Kebangsaan Malaysia, Kuala Lumpur, Malaysia; Discipline of Medical Imaging, Department of Imaging and Applied Physics, Curtin University, GPO Box U1987 Perth, Western Australia, Australia. August 2014.

[7] Cedars-Sinai, "Cerebral Angiography", *Computed Tomogrpahy (CT) Scans*. Cedars-Sinai.

[8] F. Giangregorio, A. Bertone, L. Fanigliulo, G. Comparato, G. Aragona, M.G. Marinone, G. Sbolli, P. Tansini, and F. Fornaria, "Predictive value of time–intensity curves obtained with contrast-enhanced ultrasonography (CEUS) in the follow-up of 30 patients with Crohn's disease," Gastroenterology Operative Unit, Hepatology and Digestive Endoscopy, Guglielmo da Saliceto Hospital, Piacenza, Italy. Operative Unit for Gastroenterology and Digestive Endoscopy, San Giovanni Vecchio Hospital, Turin, Italy. November 2009.

[9] Yannan Yu, Danfeng Guo, Min Lou, David Liebeskind, Fabien Scalzo, "Prediction of Hemorrhagic Transformation Severity in Acute Stroke From Source Perfusion MRI," IEEE. December 2017.

[10] Noah Stier, Nicholas Vincent, David Liebeskind, Fabien Scalzo, "Deep learning of tissue fate features in acute ischemic stroke," IEEE. November 2015.

[11] National Electrical Manufacturers Association (NEMA), "DICOM Overview" DICOM Standards Committee. 1993-2019.

[12] Merriam-Webster, "voxel", Merriam-Webster Dictionary.

[13] Benji Brandt, Roy Lin, David Macaraeg, "Tissue Fate Prediction in Acute Stroke based on MRI", GitHub. June 2019.

[14] Pixmeo, "OsiriX Viewer", Pixmeo SARL, January 2013.

[15] The Horos Project, "Horos", The Horus Project.

[16] Kasper Marstal, "Simple Elastix: Medical Image Registration Library", GitHub. January 2015.

[17] Google Cloud, "Google Cloud Platform", Google. April 2008.

[18] Stephanie Andale, "Normalized Data / Normalization", Statistics How To: Data Science Central. November 2015.

[19] Krishni Hewa, "K-Fold Cross Validation", Medium. December 2018.

[20] Koo Ping Shung, "Accuracy , Precision, Recall or F1?", Towards Data Science. March 2018.