One Shot Learning for Acute Kidney Injury Prediction

Harshitha Onkar
Department of Computer Science
California State University,
Sacramento, USA
harshithaonkar@csus.edu

Gargi Prabhugaonkar
Department of Computer Science
California State University,
Sacramento, USA
gprabhugaonkar@csus.edu

ABSTRACT

Acute kidney injury (AKI) is the most common postoperative complication among surgical patients. Postoperative AKI can prolong the hospitalization period and increase the risk of both in-hospital mortality and chronic kidney disease [1]. AKI is a common side effect observed in patients after liver resection. Hence early prediction of postoperative AKI would help the patients as well as medical personnel to take appropriate action to reduce mortality rate. Most of the existing prediction models are based on machine learning or regular deep learning models. These models often require either large or balanced datasets to make good predictions. Large amount of publicly available medical datasets is hard to find, additionally they are usually imbalanced datasets. In order to handle small-sized datasets we propose one shot learning approach with Triplet Siamese Network (TSM) and downstream classifier ensemble using deep learning models to produce better AKI predictions. The Triplet Network was first implemented in FaceNet Paper [2]. These models are widely used for image similarity detection using anchor, positive and negative triplets. We propose on applying similar approach on AKI numerical data. A Triplet Siamese Network (TSN) is trained on Anchor, Positive, Negative triplets to learn correlation among them and produce compact embedding vector using the triplet-based loss function [3]. The triplet loss function is defined such that the positive samples are grouped closer to the anchor sample and the negative samples are placed far apart from the anchor in the higher dimensional space. Such embeddings when given as input to downstream classifier model produces better classification results as opposed to raw data. Our experiments on AKI dataset [4] shows improved Receiver Operating Characteristic curve and per label F1 score results (especially for the AKI minority class) in comparison to machine learning model implementation in the base paper [1].

KEYWORDS

Triplet Loss; Siamese Network, Triplet Siamese Model (TSM); Acute Kidney Injury Prediction (AKI); One-shot learning, Fully Connected Neural Network (FCNN), Convolutional Neural Network (CNN).

1. INTRODUCTION

Acute kidney injury (AKI), a condition characterized by persistent oliguria and elevated serum creatinine levels, is a common complication in patients undergoing surgery [5]. The incidence of postoperative AKI accounts for 18%_47% of total hospitalized AKI patients. AKI is a common side effect observed in patients after liver resection [1]. Early prediction of AKI can be beneficial to patients as well as doctors as timely actions can be facilitated to save several lives affected by AKI. Although the exact mechanism of AKI is not yet fully understood, loss of homeostasis in the immune system and the ensuing inflammatory response is now believed to play major roles in the development of AKI [4].

Many studies have used classical regression methods to identify risk factors and construct risk prediction models. However, compared with conventional analysis methods, machine learning techniques minimize these limitations and may perform better. Studies have shown that machine learning can predict AKI after liver transplant, cardiac surgery, severe burns and percutaneous coronary intervention [1].

A study investigated the preoperative risk factors associated with secondary AKI after hepatectomy. It used machine learning techniques (logistic regression, decision tree and Gradient Boosting) to construct a predictive model of secondary AKI after hepatectomy, thus providing guidance for clinical therapies, and improving surgical patient prognosis [1].

Deep learning is one of the most popular research fields at present, and it has been widely used in many domains of science and proved to be very effective, especially in the predictions and classifications of objects. In recent years, deep learning methods have been applied to develop a classification model according to historical data (trainset) and utilized this model to recognize on test data (test set). However, a conventional deep learning method generally requires that there is enough training data for establishing a prediction model [6].

Classification under the restriction that we may only observe a single example of each possible class before making a prediction about a test instance. This is called one-shot learning [7]. One-shot learning is one of the ways of learning good similarity measure [8].

In our work, we aim to predict postoperative AKI after liver cancer resection. Publicly available medical datasets have limited access and require additional training to handle patient sensitive data appropriately. Therefore, we focus our approach on one-shot learning and more specifically on Triplet Siamese Model (TSM), which adds Triple Loss to a Siamese Network so that accurate predictions can be made using a small set of data. We implement a Triplet Siamese Model to predict postoperative AKI after liver cancer resection. For AKI predictions, we implement deep learning downstream models. Additionally, we compare our results with previous work to prove that deep learning methods yield accurate prediction results when combined with TSM. Currently Siamese Networks are widely used with image data. We also prove through our implementation that TSM can be extended to numerical data.

To summarize, the key contributions of our work are:

- Integration of Triplet Siamese Model with deep learning classifiers for AKI prediction.
- Implementation of Triplet Siamese Model for one-shot learning on numerical data.
- Implementation of Deep Learning Classification Models- Convolutional Neural Network and Fully Connected Neural Network for AKI prediction.
- Implementation of performance metrics of previous work on machine learning approach for AKI prediction for comparison purpose.

The remainder of this paper is organized as follows: Related work is described in Section 2, Problem Formulation of our work is described in Section 3, System/Algorithmic Design which includes an overview of the implemented ensemble, algorithmic flow, system modules have been explained in detail in Section 4, Experimental Evaluation, which includes methodology, experimental setting, and results and comparison of Machine Learning approach with One-shot learning with deep neural network classification have been described in Section 5. Finally, Conclusion, Work Division, Learning Experience, Acknowledgement have been stated in Sections 6-9.

2. RELATED WORK

Machine learning methods have been widely used to predict the likelihood of acute kidney injury after liver cancer resection [1]. This work has identified the important parameters and have drawn a conclusion that Machine learning technology can predict acute kidney injury after hepatectomy. Parameters such as Age, cholesterol, tumor size, surgery duration and PLT influence the likelihood and development of postoperative acute kidney injury [1].

Conventional deep learning method generally requires that there is enough training data for establishing a prediction model Few-shot learning approach has been used in Software Defect Prediction (SDP) to classify software defect modules into defect and non-defect [6]. Humans are very good at identifying objects with little direct supervision or none at all, called few-shot learning. Due to the inspiration from the few-shot learning ability of humans, there has been the renewed interest in few-shot learning recently and the Siamese networks for few-shot learning where a little data is available were proposed [6].

Previous work on verification, recognition and clustering has been done using image data. The method is based on learning a Euclidean embedding per image using a deep convolutional network. FaceNet directly trains its output to be a compact 128-D embedding using a triplet-based loss function [2]. In this work, the triplets consist of two matching face thumbnails and a non-matching face thumbnail and the loss aims to separate the positive

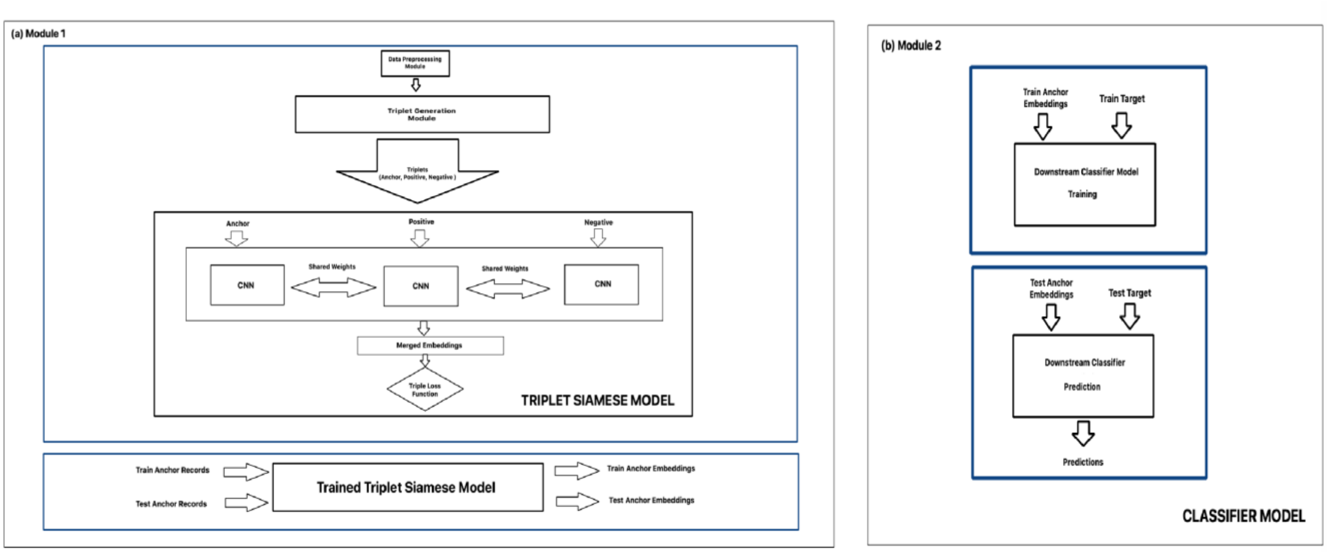


Figure 1: Overview of system architecture (a) Module 1 (b) Module 2

pair from the negative by a distance margin [2].

In our work, we extend the idea of using a unified system for verification (identifying similarities), recognition (identifying categories) and clustering (grouping similar categories) AKI data. We use the idea of generating triplets which consists of two matching pairs and one non-matching pair as used in FaceNet to generate AKI positive and AKI negative pairs. Similarly, we train our embedding model to generate a compact embedding vector using the triplet-based loss function. We then use a downstream classifier model to make predictions about AKI.

In our work, we use the dataset used in [1] to train the upstream embedding model of our unified system. Next, using the same dataset, we generate train and test embedding for all records. By using a downstream classifier model, we are able to accurately predict postoperative AKI after liver cancer resection. Our work reflects the use of TSM on numerical dataset to generate embeddings that can be later used for classification problems. This approach eliminates the need of feature importance analysis as done in [1] and also prove that Triplet Loss and Siamese Network does not limit to image data. It can be extended to numerical data for accurate predictions using neural networks. We have also identified that one-shot learning similar to that used in [6], can be extended to other domains such as medicine where most data is not public and thus, not much data is available for model training and testing purpose.

By building a Triplet Siamese model for one-shot learning and a downstream classifier model, we are able to accurately perform AKI prediction.

3. PROBLEM FORMULATION

The dataset is numerical in nature and consists of patient's data such as age, intraoperative, laboratory and postoperative data as input features. The target column is Kidney Disease: Improving Global Outcomes (KDIGO), which defines postoperative AKI as an increase in serum creatinine ≥0.3 mg/dL within 2 days after surgery, or an increase ≥1.5-fold in serum creatinine within 7 days after surgery [9].

The proposed ensemble consists of a Triplet Siamese Network and a downstream classifier model. The TSM takes Anchor (A), Positive (P) and Negative (N) triplets as input and generates embeddings as output. The downstream model is trained on the anchor embeddings to output a class (in this case AKI or Non-AKI). The APN triplets are selected in such a way that Positive belongs to same class as Anchor and Negative belongs to different class than Anchor. The loss function for the Triplet Network is defined using these embeddings as:

$$L(A, P, N) = \max(\|f(A) - f(P)\|^2 - \|f(A) - f(N)\|^2 + \alpha, 0)$$

where A is an anchor input, P is a positive input of the same class as A, N is a negative input of a different class from A, α is a margin between positive and negative pairs, and function f represents an embedding [10] [3].

4. SYSTEM/ALGORITHM DESIGN

4.1. System Architecture

In this section, we describe our system which consists of two modules. Figure 1 shows an overview of the system. Module 1 (see item (a) in Figure 1) consists of sub-modules such as data preprocessing module, triplet-generation module, and the Triplet Siamese Model (TSM). After the TSM is trained, it is used to generate anchor embeddings for the train set and test set. The embeddings from module 1 are then used in Module 2 (see item (b) in Figure 1) by the downstream model. Module 2 describes downstream model training and prediction. For the classifier model training, the train anchor embeddings are used along with the train output channel (AKI or Non-AKI). For the classifier model prediction, the test anchor embeddings are used along with the test output class to predict if a patient has an AKI condition.

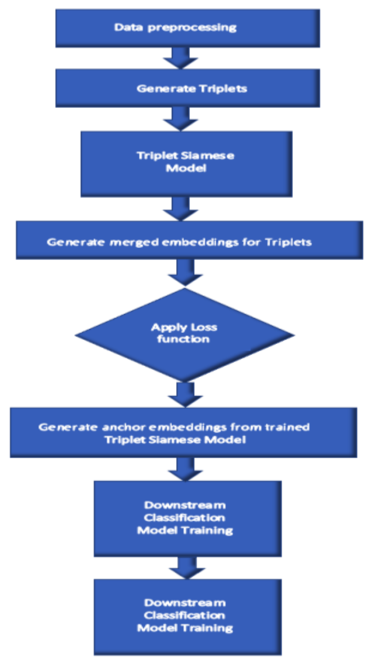


Figure 2: Algorithmic Flow-chart of the ensemble

We also describe the flow of data and control between the different modules of the system in Figure 2. It identifies the stepwise processing of data from the data preprocessing step to the downstream model prediction. It describes how Module 1 and Module 2 integrate to generate prediction results.

For Triplet Siamese model, we have used Convolutional Neural Network (CNN) and for downstream model, we have compared our results using a Fully Connected Neural Network (FCNN) and CNN. The dataset used in our work is unbalanced as it contains fewer AKI data compared to non-AKI data. Therefore, in our second approach described in Section 5.2, we have used the up-sampling technique in Module 1(see Figure 2) after data preprocessing module and before triplet generation module to obtained much better results.

4.2. Data Preprocessing

For data preprocessing the following steps are carried out:

- Dataset was checked for null values
- Numerical data was normalized using z-score
- Categorical columns were one-hot encoded

4.3. Triplet Generation

As discussed in problem formulation section (see Section 3), triplets are selected such that Anchor (A) and Positive (P) belong to same class and Anchor and Negative belong to different classes. There are different methods by which triplets can be generated, such as offline and online triplet mining methods. Offline triplet mining involves generating triplets at the beginning of each epoch beforehand whereas online triplet mining involves computing useful triplets on the fly [11]. For our proposed system we have used two variants of offline triplet mining:

- Random combination of A, P and N
- By combining each AP pair in dataset of every class (AKI, non-AKI) with every N pair

The second triplet generation approach produced better end results compared to the random triplet generation method.

4.4. Triplet Siamese Model

The Triplet Siamese Model consists of anchor, positive, negative channels, an embedding model (shared weights channel), a concatenation layer (to concatenate APN embeddings for triplet loss evaluation) and compiled on triplet loss function. (see item (a) of Figure 1). The goal of the triplet network is to generate anchor, positive and negative embeddings in higher dimensional space such that the positive samples are closer to the anchor sample and the negative samples are far away from the anchor [12]. This is achieved using triplet loss function [2].

In our proposed system we have used Convolutional Neural Network (CNN) as embedding model (shared weights channel) (see item (a) in Figure1). The last layer of embedding model performs L2 normalization, this normalizes the output vector and maps it to the surface of n-dimensional hyper-sphere of radius one and helps in better convergence [13]. Each triplet component (anchor, positive and negative) is sent through the same shared weight channel to obtain respective embeddings. These embeddings are then merged into one single embedding along an axis using concatenate layer. The combined embedding is then used by triplet loss function to calculate the loss for the Triplet Siamese model.

The triplet loss function is the most important part of TSM as this function allows the model to learn similar class records from dissimilar classes. The triplet loss can be defined by using either Cosine functionality or Euclidian distance to compare output embeddings and understand the similarity/dissimilarity between them [13].

After training TSM, the model weights are saved.

4.5. Trained Model for Anchor Embeddings

The trained model of previous step (see item (a) of Figure 1 and Section 4.4) is used to get embeddings for train and test set. This time only the anchor input channel is used, and the trained model produces embeddings from single input channel. The train embeddings are used to train the downstream classifier model (see item (b) of Figure 1) and prediction is done using the test embeddings.

4.6. Downstream Classifier Model

The downstream classifier model (see item (b) of Figure 1) takes embeddings generated by the trained model as input and predicts the class as the output i.e. in the case 0 (non-AKI) or 1 (AKI). The downstream can be any type of classification model ranging from traditional machine learning models to deep learning models. In our proposed system we use CNN and FCNN as two types of downstream classification model.

The entire algorithmic design for the proposed system is summarized in the flow chart as shown in Figure 2.

5. EXPERIMENTAL EVALUATION

5.1 Methodology

In order to evaluate the proposed system, we used Acute Kidney Injury (AKI) after liver cancer resection dataset [14]. The dataset consists of 1173 patient records, out of which 77 (6.6%) had AKI and 1,096 (93.4%) who did not [1]. Each record has 21 columns – 20 input feature columns and a target column (KDIGO). The dataset is used to create triplets – Anchor, Positive and Negative based on target column. The triplets consist of either (AKI, AKI, non-AKI) or (non-AKI, non-AKI, AKI) as Anchor, Positive and Negative respectively.

The triplet batch is split into train-test pair in 70-30 ratio. These triplets are used to train the Triplet Siamese network compiled on triplet loss function. The trained triplet model is then used to obtain embeddings from just the anchor input channel (only anchor embeddings). The anchor embeddings are used for downstream model training and prediction. The downstream model is a classification model that predicts whether a patient has AKI or not. Since it is a classification problem, we use precision, recall, F-measure, Confusion Matrix and Receiver Operating Characteristic (ROC) curves to evaluate model performance. The performance of machine learning models in the base paper and the proposed system are compared on ROC curves and per label F1 scores.

5.2 Experimental Setting

The experiment is carried out using two different approaches based on variations in embedding model architecture, embedding size and triplet generation. The details are as described below.

5.2.1 Approach 1

For this approach the architecture of CNN embedding channel and the downstream classifier model FCNN and CNN is as described in Table 1, 2 and 3 respectively. The TSM model summary as described in Section 4.4 is provided in Figure 3. A variation of Offline Triplet generation [15] was used to generate triplets. The no. of train and test triplets generated were 7800 and 3450 respectively. In this approach, after data preprocessing and normalization, we identified 35 features which is used as the embedding dimension as well as input shape to downstream classifier models. The other hyperparameter settings for the upstream and downstream models are given below.

Triplet Siamese Network (Upstream Model)

- Alpha/Margin: 0.8
- Embedding Dimension: 35
- Optimizer: Custom Adam with lr = 0.0001, beta_1 = 0.9 and beta_2=0.999, epsilon= 1e-08
- Total Trainable Parameters: 4,333,603

FCNN Classifier (Downstream Model Type 1)

- Input shape: 35
- Optimizer: Default Adam
- Total Trainable Parameters: 12,994

CNN Classifier (Downstream Model Type 2)

- Input shape: 1 x 35 x 1
- Optimizer: Default Adam
- Total Trainable Parameters: 6,073,090

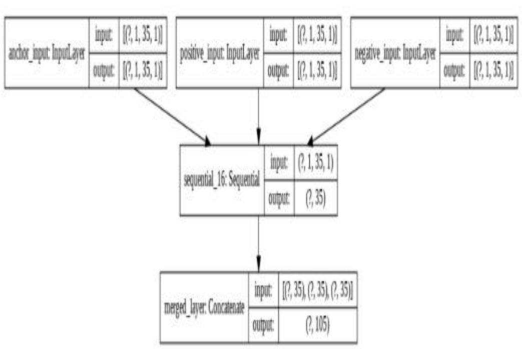


Figure 3. Approach 1- Triplet Siamese Model

Layers	Activation	Kernel/Stride	Input Size
Conv1	ReLU	(7x5) x 512 /1	1x35x1
MaxPool1		(2x2) /2	1x35x512
Conv2	ReLU	(5x5) x256 /1	1x18x512
MaxPool2		(2x2) /2	1x18x256
Conv3	ReLU	(5x5) x128 /1	1x9x256
MaxPool3		(2x2) /2	1x9x128
Conv4	ReLU	(5x5) x64 /1	1x5x128
MaxPool4		(2x2) /2	1x5x64
Flatten			1x3x64
Dense	ReLU	35	35
L2 Norm			35

Table 1. The structure of CNN embedding model for Approach 1. For the convolution layer, the kernel is specified as (m×m) sized filter × (# of filters) / (# of stride). For the max-pooling layer, (p×p) sized pooling windows / # of stride. The input sizes are denoted as rows × cols × (# of filters)

Layers	Activation	Input Size
Input		35
Dense1	ReLU	35
Dense2	ReLU	128
Dense3	Softmax	64

Table 2. The structure of Fully connected Neural Network Classifier model used in Approach 1.

Layers	Activation	Kernel/Stride	Input Size
Input			1x35x1
Conv1	ReLU	(1x35) x512 /1	1x35x1
MaxPool2		(1x2) /1	1x35x512
Conv2	ReLU	(1x35) x 256 /1	1x18x512
MaxPool2		(1x2) /1	1x18x256
Conv3	ReLU	(1x35) x 128 /1	1x9x256
MaxPool3		(1x2) /1	1x9x128
Conv4	ReLU	(1x35) x 64 /1	1x5x128
MaxPool3		(1x2) /1	1x5x64
Flatten			1x3x64
Dense1	ReLU	128	192
Dense2	ReLU	64	128
Dense3	Softmax	2	64

Table 3. The structure of Downstream CNN Classifier model used in Approach 1. For the convolution layer, the kernel is specified as (m×m) sized filter × (# of filters) / (# of stride). For the max-pooling layer, (p×p) sized pooling windows / # of stride. The input sizes are denoted as rows × cols × (# of filters)

5.2.2 Approach 2

For this approach the architecture of CNN embedding channel and the downstream classifier model architecture is as described in

Table 4 and 5 respectively. The TSM model summary as described in Section 4.4 is provided in Figure 4. A variation of Offline Triplet generation [15] is used to generate triplets. In this approach, after data preprocessing and normalization, we identified 23 features. The data used to generate the triplets is up sampled using Synthetic Minority Oversampling Technique (SMOTE) [16]. The no. of train and test triplets generated after oversampling are 56000 and 24000 respectively. The other hyperparameter settings for the upstream and downstream models are as below.

Triplet Siamese Network (Upstream Model)

- Alpha/Margin: 0.8
- Embedding Dimension: 128
- Optimizer: Custom Adam with lr = 0.0001, beta_1 = 0.9 and beta_2=0.999
- Total Trainable Parameters: 74,752

CNN Classifier (Downstream Model)

- Input shape: 16 x 8 x 1
- Optimizer: Default Adam
- Total Trainable Parameters: 52,098

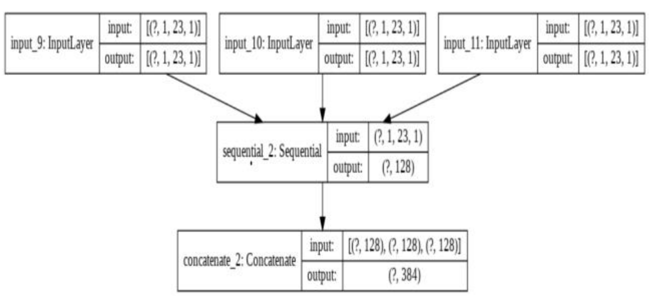


Figure 4. Approach 2- Triplet Siamese Model

Layers	Activation	Kernel/Stride	Input Size
Conv1	ReLU	(1x5) x 64 /1	1x23x1
BatchNorm1			1x19x64
MaxPool1		(1x2) /1	1x19x64
Dropout			1x10x64
Conv2	ReLU	(1x5) x128 /1	1x10x64
BatchNorm2			1x6x128
MaxPool2		(1x4) /1	1x6x128
Dropout2			1x2x128
Flatten			1x2x128
Dense	ReLU	128	256
L2 Norm			128

Table 4. The structure of ConvNet embedding model for Approach 2. For the convolution layer, the kernel is specified as (m×m) sized filter × (# of filters) / (# of stride). For the max-pooling layer, (p×p) sized pooling windows / # of stride. The input sizes are denoted as rows × cols × (# of filters)

Layers	Activation	Kernel/Stride	Input Size
Input			16x8x1
Conv1	ReLU	(2x2) x128 /1	16x8x1
MaxPool1		(1x2) /1	15x7x128
Conv2	ReLU	(2x2) x 64 /1	15x3x128
MaxPool2		(1x2) /1	14x2x64
Conv3	ReLU	(2x1) x 32 /1	14x1x64
MaxPool3		(2x1) /1	13x1x32
Flatten			6x1x32
Dense1	ReLU	64	192
Dense2	ReLU	32	64
Dense3	Softmax	2	32

Table 5. The structure of Downstream CNN Classifier model used in Approach 2. For the convolution layer, the kernel is specified as (m×m) sized filter × (# of filters) / (# of stride). For the max-pooling layer, (p×p) sized pooling windows / # of stride. The input sizes are denoted as rows × cols × (# of filters)

5.3 Results

We observe the distribution of AKI and non-AKI data by using t-distributed Stochastic Neighbor Embedding (TSNE). It is a tool which is used to visualize high-dimensional data. It converts similarities between data points to joint probabilities and tries to minimize the Kullback-Leibler divergence between the joint probabilities of low-dimensional embedding and high dimensional data [17]. We used a scatter-plot function to visualize the clustering of data before and after the application of Triplet Siamese Model (see Figure 5,6,8,9). We present our observations and results from the two approaches below.

We also reproduce results from one of the references [1] wherein machine learning algorithms have been used for AKI prediction. In the end, we compare our results with the outcomes of machine learning algorithms.

5.3.1 Approach 1

In our first approach, without using oversampling technique, we implement TSM with a variation of triplet generation technique and loss function. We use TSNE tool for data visualization to observe the data point distribution of AKI and non-AKI data before and after TSM implementation. The data points for AKI data are indicated in red and data points for non- AKI are indicated in blue. As can be seen in Figure 5 (see item (a) for train set data and (b) for test set data), the data points for AKI and non-AKI are distributed across. Therefore, correlation between the data is not established. By training a FCNN on this dataset does not yield good prediction results.

However, after applying Triplet Siamese Model and generating embeddings we observe clustering of data points such that the

AKI embeddings are clustered together as can be seen in Figure 6 (see item (a) for train set data and (b) for test set data).

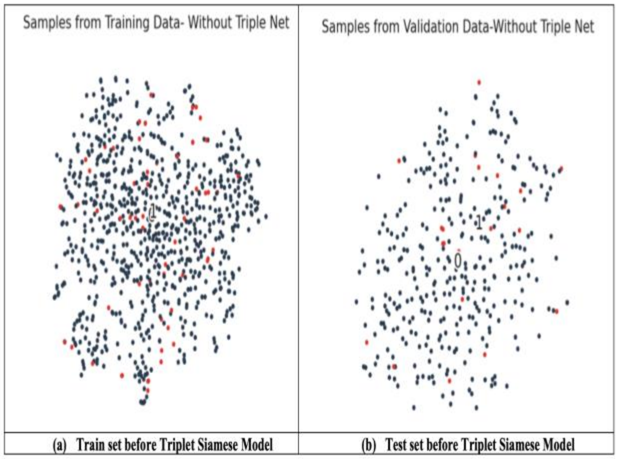


Figure 5. Approach 1- Train and Test set without Triple Siamese Model

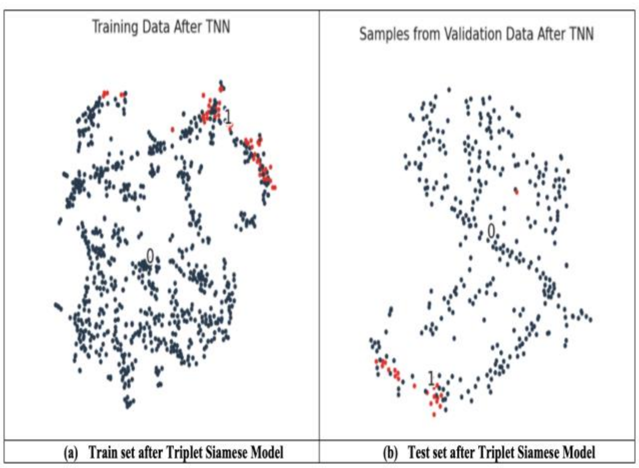


Figure 6. Approach 1- Train and Test set with Triple Siamese Model

The embeddings generated from TSM are then used to train a downstream classifier model. This model is used for the purpose of AKI prediction. In approach 1, we observe the predictions generated by two types of classifier models- (a) a Fully Connected Neural Network (FCNN), and (b) Convolutional Neural Network (CNN). The performance metrics of the two models are presented in Figure 7 (see item (a) for FCNN (b) CNN). The performance metrics include the scores for accuracy, precision, recall and F1. Additionally, the confusion matrix indicating True-Positive (TP), True-Negative (TN), False-Positive (FP) and False-Negative (FN) values has been included. The ROC plot has been included for reference.

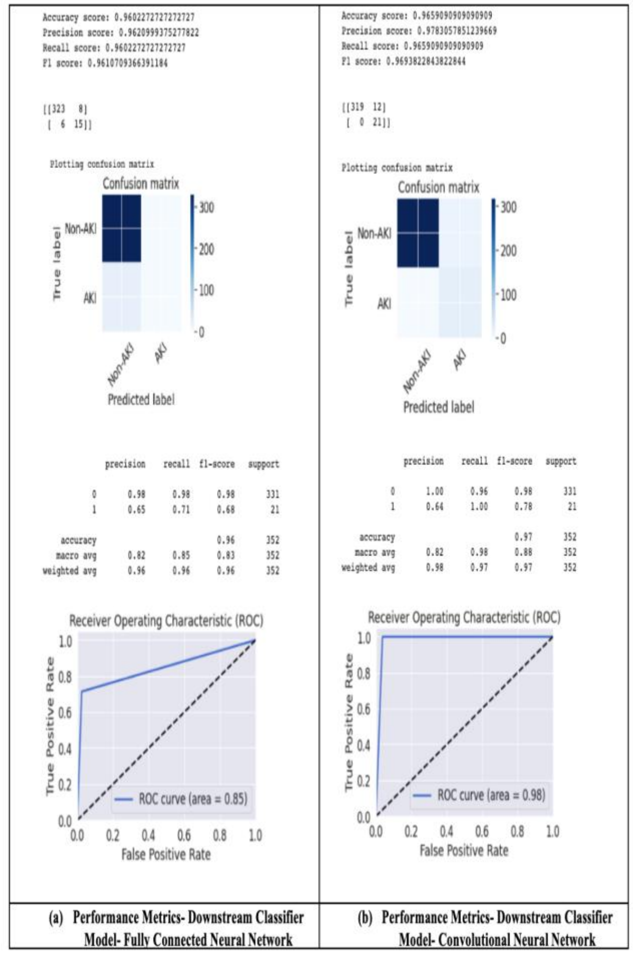


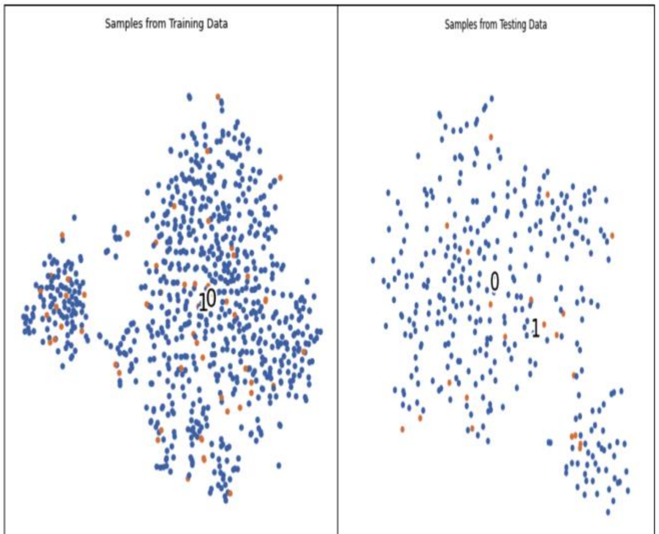
Figure 7. Approach 1- Performance Metrics with downstream model as Fully Connected Neural Network and Convolutional Neural Network

In Figure 7, we observe that the CNN classifier model performs better than FCNN classifier model as it is able to accurately classify all the AKI records. Although the F1 score for non-AKI labels are equal (0.98), the CNN model has a higher F1 score for AKI label (0.78) compared to that for FCNN (0.68). Comparing the ROC plots for the two models, it can be seen that CNN has a higher Area under ROC Curve (AUC) (0.98) compared to FCNN (0.85). We thus conclude that CNN is a better model for TSM implementation using numerical data as well as for downstream classification for AKI prediction.

5.3.2 Approach 2

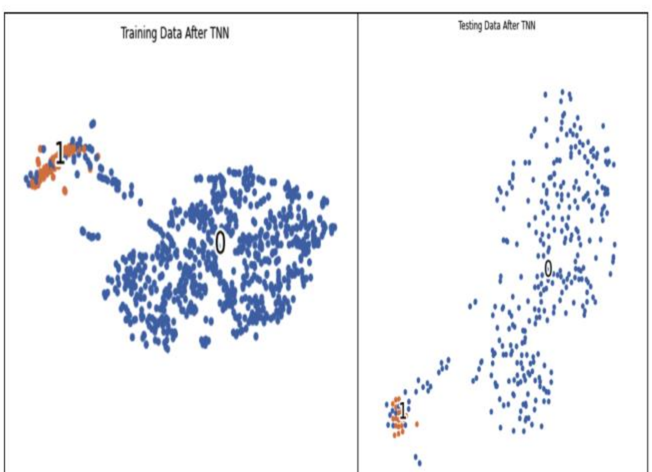
In our second approach, we use oversampling technique. We use TSNE tool for data visualization to observe the data point distribution of AKI and non-AKI data before and after Triplet Siamese Model implementation. The data points for AKI data are indicated in orange and data points for non- AKI are indicated in blue. As can be seen in Figure 8 (see item (a) for train set data and

(b) for test set data), the data points for AKI and non-AKI are scattered throughout. In order to learn similar and dissimilar class records the Triplet Siamese Network is used. After training the triplets on the above model when the embeddings are plotted again, similar class records can be seen to form clusters as shown Figure 9 (see item (a) for train set data and (b) for test set data). The embeddings from the trained model are given as input to downstream classifier model. In Approach 2, we use CNN as downstream model. The performance metrics of the approach 2 ensemble is shown in Figure 9.



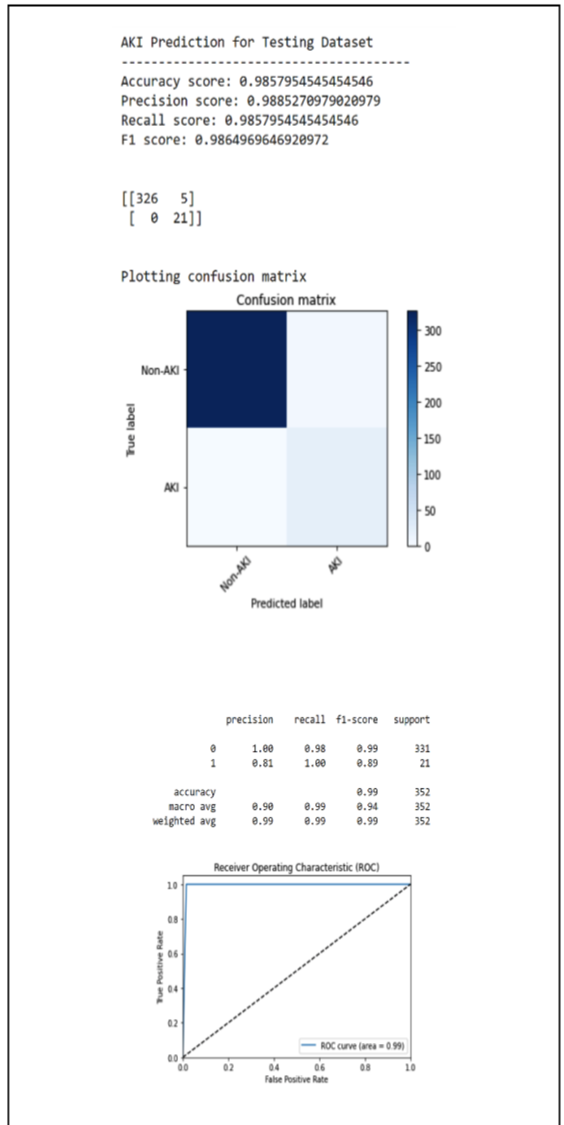
(a) Train set before Triplet Siamese Model (b) Test set before Triplet Siamese Model

Figure 8. Approach 2- Train and Test set without Triple Siamese Model



(a) Train set after Triplet Siamese Model (b) Test set after Triplet Siamese Model

Figure 9. Approach 2- Train and Test set with Triple Siamese Model



(a) Performance Metrics- Downstream Classifier Model- Convolutional Neural Network

Figure 10. Approach 2- Performance Metrics with downstream model as Convolutional Neural Network

5.3.3 Machine Learning for AKI prediction

Our work is inspired from one of the related works [1] that used machine learning methods for AKI prediction. We use this work as a reference, to build a Triplet Siamese Model for generating embeddings that are used in the downstream classification model. We have used the same dataset as used for AKI predictions using machine learning algorithms (ML) in the reference work. Therefore, we can compare our results with the results of ML models. We have recreated the results using the code available and shared by the authors. Additionally, for the purpose of comparison, we have re-implemented the ML algorithms using the publicly available code in [1] and have also generated performance metrics for the ML algorithms on similar parameters used in our work. We have used the same function to generate the performance scores, confusion matrix and ROC plots.

In the reference work [1], feature importance analysis has been performed using Gbdt algorithm as can be seen in Figure 11. Their results indicate that age, cholesterol, tumor size, surgery duration and PLT are the most important features among the total 21 features available in this dataset.

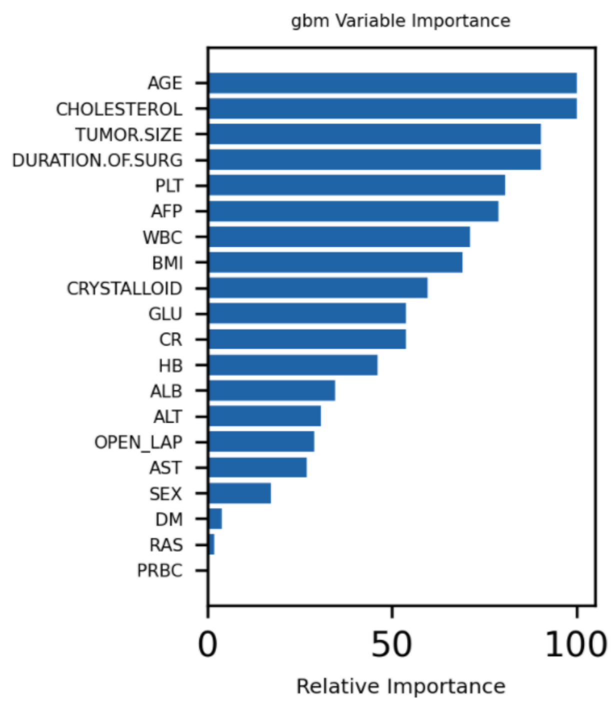


Figure 11. Feature importance analysis using Gbdt algorithm

We have generated performance metrics for all the ML algorithms used in [1] , namely Logistic Regression, Decision Tree, Random Forest, Gradient Boosting Decision Tree (Gbdt), XGBoost Classifier and Gradient Boosting Machines (GBM). More details on these algorithms and the work on AKI predictions is available in the reference paper [1].

For the purpose of comparison of the Confusion Matrix and ROC plot, we have included only Random Forest Classifier and GBM Classifier in Figure 12 (see item (a) Random Forest Classifier, and (b) GBM Classifier). The evaluation indicates that machine learning algorithms can successfully classify non-AKI data with higher F1 score (0.99) but does not fully classify the AKI data. As can be seen out of the 23 AKI records, only 2 records have been correctly classified as AKI. The AUC for all the classification models used in [1] can be seen in Figure 13. It is evident that Random Forest and Gbdt have higher AUC compared to the other machine learning algorithm used in this work.

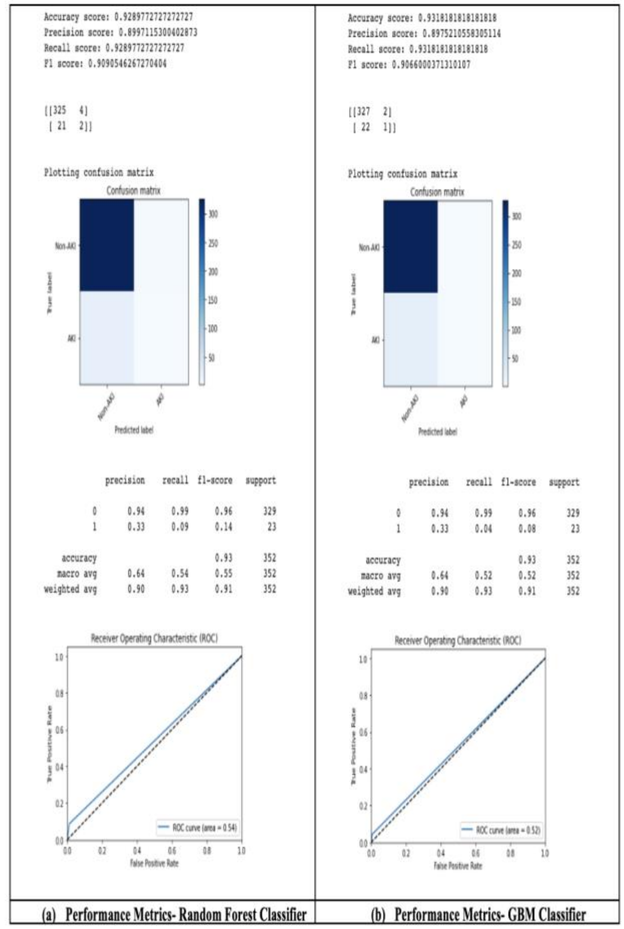


Figure 12. Performance Metrics- Machine learning algorithm

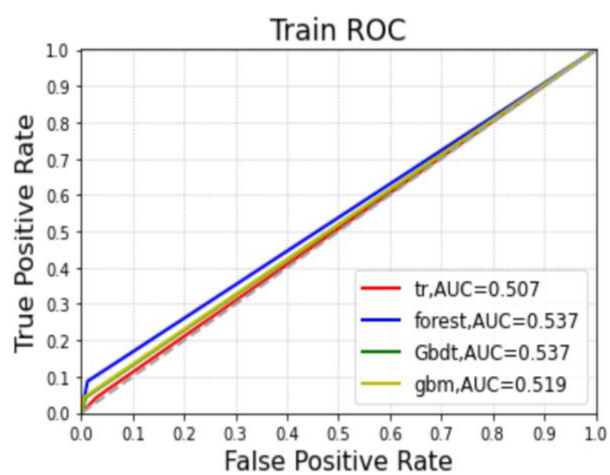


Figure 13. ROC plots- Machine learning algorithms [1]

5.3.4 Comparison of Machine Learning with Integrated Triplet Siamese with Deep Learning Classification

In this section we compare the results of the reference paper [1], with our work. We compare the FI scores for the two classification classes- Class 0 for Non-AKI and Class 1 for AKI. We present our comparison in Table 6.

As can be observed, compared to machine learning approach, our work on using one-shot learning using Triplet Siamese Model and Deep Learning Classifier model yields better results. The best performing machine learning approach- Random Forest Classifier achieves an F1 score of 0.96 and 0.14 for Class 0 and Class 1 respectively. Our best performing approach of using oversampling on the same unbalanced data set achieves a F1 score of 0.99 and 0.89 for Class 0 and Class 1 respectively. Additionally, our work successfully classifies all the AKI records and a majority of non-AKI records correctly.

Our work indicates that one-shot learning using Triplet Siamese Model eliminates the need for feature important analysis to be done separately. Moreover, by just using an upstream one-shot learning model and a downstream classification model, we are successfully able to develop an efficient classification system that can accurately classify AKI and non-AKI data.

Model	**Non-AKI (Class 0)**	**AKI (Class 1)**
Machine Learning Model Performance		
Decision Tree	0.95	0.06
Random Forest	0.96	0.14
GBM	0.96	0.08
Gbdt	0.96	0.07
TSM + Deep Learning Classifier System Performance		
Approach 1 - FCNN	0.98	0.68
Approach 1 - CNN	0.98	0.78
Approach 2 - CNN	0.99	0.89

Table 6. Comparison of Machine Learning Model Performance with Triple Siamese Model (TSM) + Deep Learning Classifier System.

6. CONCLUSION

In this paper, we applied One-shot learning approach using Triplet Siamese Model. Triplet Loss and Siamese network are generally used for image similarity/dissimilarity detection. Through our work, we prove that it can be extended to a numerical dataset. Our experiment on small-sized medical dataset showed encouraging results in terms of AUC and F1 scores. The implemented ensemble extends the application of One-Shot learning to non-image dataset and thus provides an advantage of applying it to medical domain where datasets are generally small, unbalanced and not easily accessible.

7. WORK DIVISION

Team Member	
Harshitha	Research & Analysis, Base Paper Performance Metrics, Approach 2- Data preprocessing, Triplet generation, Triple Loss function, Triple Siamese Model implementation, Triplet embedding visualization, downstream CNN Model, Performance Metrics, Documentation
Gargi	Research & Analysis, Base Paper re-implementation, Approach 1- Data preprocessing, Triplet generation, Triple Loss function, Triple Siamese Model implementation, Triplet embedding visualization, downstream FCNN, CNN Model, Performance Metrics, Documentation

Table 7. Task division

8. LEARNING EXPERIENCE

The project helped us to become familiar with the various terminologies associated with One shot learning approach such as Triplet generation, Triplet Siamese Network, Embedding model, Triplet loss function. It provided us an opportunity to apply One Shot learning to a new domain and dataset type. We learnt triplet generation using offline triplet mining and how to define and compile TensorFlow model with custom loss function.

The project also gave us an opportunity to develop an AI ensemble consisting of more than one model and applying the output of one model as input to another downstream model. It helped us to build on concepts that we learnt throughout our course such as data preprocessing, deep learning models (CNN, FCNN), early stopping and hyperparameter tuning and extend these concepts to newer model implementation- Triplet Siamese Network.

As the work of using Triple Loss and Siamese Network on numerical dataset is very limited, through our work we were able to learn how to leverage existing work and extend it to a new domain and data type. Starting from idea formulation to implementing the visualized system, we learnt how to apply our knowledge in AI in the medical domain.

9. ACKNOWLEDGMENT

We are thankful to Dr. Chen for providing this learning opportunity and taking time for guiding us throughout the project. His feedback and inputs have been indispensable for the completion of this project.

10. REFERENCES

[1] L. Lei, Y. Wang, Q. Xue, J. Tong, C. Zhou and J. Yang, "A comparative study of machine learning algorithms for predicting acute kidney injury after liver cancer resection," *PeerJ,* pp. 8, e8583., 2020.

[2] F. K. D. &. P. J. Schroff, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015.

[3] D. K. J. P. Florian Schroff, "FaceNet: A unified embedding for face recognition and clustering," [Online]. Available: https://arxiv.org/abs/1503.03832.

[4] Y. Moon, I. Jun, K. Kim, S. Kim, J. Song and G. Hwang, "Moon, Y. J., Jun, I. G., Kim, K. H., Kim, S. O., Song, J. G., & Hwang, G. S. (2017). Comparison of acute kidney injury between open and laparoscopic liver resection: propensity score analysis.," *PLoS One,* vol. 12, no. 10, p. e0186336, 2017.

[5] J. T. Park, "Postoperative acute kidney injury," *Korean journal of anesthesiology* , pp. 70(3), 258., 2017.

[6] L. S. Z. Z. L. Q. A. &. T. Y. Y. Zhao, "Siamese dense neural network for software defect prediction with small data," *IEEE Access,* vol. 7, pp. 7663-7677, 2018.

[7] G. Z. R. &. S. R. Koch, "Siamese neural networks for one-shot image recognition," *ICML deep learning workshop,* vol. 2, 2015.

[8] D. Banerjee, "Semi-supervised Learning using Triple-Siamese Network," *(Doctoral dissertation).*, 2020.

[9] F. d. M. U. J. V. M. D. M. B. &. C. A. F. P. Teles, "Acute kidney injury in leptospirosis: the Kidney Disease Improving Global Outcomes (KDIGO) criteria and mortality," *Clinical nephrology,* vol. 86(6), p. 303, 2016.

[10] "Triplet Loss," Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Triplet_loss.

[11] O. Moindrot, "Triplet Loss and Online Triplet Mining in TensorFlow," 2018. [Online]. Available: https://omoindrot.github.io/triplet-loss.

[12] Y. W. K. J. &. T. K. A. Kwon, "Wi-Fi Based Handwritten Signature Verification Using a Triplet Network," in *In Proceedings of the 2019 3rd International Conference on Advances in Image Processing, (pp. 190-195).*, 2019.

[13] P. Gupta, "Illustrated Guide to Siamese Network using triplet loss for One-shot learning," 2020. [Online]. Available: https://towardsdatascience.com/illustrated-guide-to-siamese-network-3939da1b0c9d.

[14] I.-G. J.-H. K.-O. K.-G. S. ,.-S. H. Young-Jin Moon, "Comparison of acute kidney injury between open and laparoscopic liver resection: Propensity score analysis," 2017. [Online]. Available: https://doi.org/10.1371/journal.pone.0186336.

[15] KinWaiCheuk, "Github-Triplet-net-keras," [Online]. Available: https://github.com/KinWaiCheuk/Triplet-net-keras.

[16] N. V. B. K. W. H. L. O. &. K. W. P. Chawla, "SMOTE: synthetic minority over-sampling technique," *Journal of artificial intelligence research,* vol. 16, pp. 321-357, 2002.

[17] "sklearn.manifold.TSNE," [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html.

[18] J. V. d. M. U. D. M. B. M. a. A. F. P. C. Flávio Teles, "Acute kidney injury in leptospirosis: the Kidney Disease Improving Global Outcomes (KDIGO) criteria," *Clinical nephrology,* 2016.

[19] P. T. M. Ignatius Y Tang, "Prevention of perioperative acute renal failure: what works?," [Online]. Available: https://doi.org/10.1016/j.bpa.2003.09.006.

[20] W. Y. X. Q. T. J. Z. C. Y. J. Lei L, "A comparative study of machine learning algorithms for predicting acute kidney injury after liver cancer," [Online]. Available: https://doi.org/10.7717/peerj.8583.

[21] "Triplet_loss," [Online]. Available: https://en.wikipedia.org/wiki/Triplet_loss.

[22] I. Y. &. M. P. T. Tang, "Prevention of perioperative acute renal failure: what works?," *Best Practice & Research Clinical Anaesthesiology,* vol. 18(1), pp. 91-111, 2004.