DIAGNOSIS CARDIAC DISEASES BY ROBUST-FEATURE SELECTION FROM THE COMPLEX LATENT SPACE OF DL-BASED SEGMENTATION NETWORKS

A PREPRINT

Fahim Ahmed Zaman*

Wahidul Alam†

ABSTRACT

Researchers have shown significant correlations among segmented objects or layers in medical image modalities and pathologies. Several studies showed that using hand crafted features for disease prediction neglects the immense possibility to use deep learned features from segmentation models which may hamper the overall classification accuracy. However, directly using classification networks on medical images itself or segmented images opt out robust feature selection and may lead to overfitting and hence reduced test accuracy. To fill that gap, we propose a novel feature selection technique using the latent space of a Deep Convolutional Neural Network (DCNN) that can aid classification and prognosis. We tested our method in differentiating a rare cardiac disease: Takotsubo Syndrome (TTS) from the ST elevation myocardial infarction (STEMI). Our approach shows promising results in classifying the diseases by beating the state-of-the-art approach by 1.2%. Our approach also shows great potential in reducing the redundant features by maintaining reasonable classification accuracy and creates a robust pipeline for disease prognosis prediction in the downstream analysis.

Keywords Segmentation latent space · feature selection · feature reduction · Takotsubo Syndrome

1 Introduction

Quick diagnosis and accurate treatment by giving proper medication to patients is necessary for life threatening diseases such as acute myocardial infarction (AMI). But the Takotsubo syndrome (TTS) can mimic clinical and electrocardiographic (ECG) features of AMI and hard to distinguish between them using just echocardiogram videos (echo). Use of angiogram is the conventional way of distinguishing between these two diseases which is not only invasive, but also slow in process that may endanger the patient. Recently deep learning models are proposed to classify TTS and STEMI using echo Zaman et al. [2021], Laumer et al. [2022]. In our earlier study, we demonstrated that using deep learning models as binary classifier between the two diseases can significantly improve the detection accuracy compared to the physicians and help them make the judgement calls in difficult cases Zaman et al. [2021]. Looking at the feature maps of the trained classifier, we identified that the basal septal, antero-lateral walls and the apex of the heart are extremely important in decision making of the deep learning models. Despite having good classification accuracy, DCNN classifiers do not guarantee robust feature selection, particularly in a noisy dataset such as echo. Moreover, artifacts and speckle noise in the echos can generate irrelevant and wrong features that may hamper the overall accuracy. Looking at the feature maps of the trained classifier, we identified that the basal septal, antero-lateral walls and the apex of the heart are extremely important in decision making of the deep learning models, but not consistent in all the individual cases as they can be affected by echo artifacts. We further investigated and found out that there are indeed significant motion differences in the above mentioned regions of the heart between the two diseases. The pathological evidences also suggest that the motion of the Left Ventricle (LV) plays a vital role in differentiating these two diseases.

*Fahim Ahmed Zaman is with the department of Electrical and Computer engineering, University of Iowa. email: fahim-zaman@uiowa.edu

†Wahidul Alam is with the department of Biomedical engineering, University of Iowa. email: mohammadwahidul-alam@uiowa.edu

But it is impossible to distinguish the spatial and temporal features of the classifier due to the black box nature of the model. This inspired us to use a segmentation network to segment LV first, then use the inherent features of the segmented LV itself to train a disease classifier, rather than using a direct DCNN classifier with echo as input. Intuitively, the segmentation related features are more robust and correlate the diseases of interest.

Baek et al., showed that CNNs trained to perform tumor segmentation task, with no other information than the physician contours, identify a rich set of survival-related image features with remarkable prognostication in a retrospective setting Baek et al. [2019]. They were able to identify strong correlation between segmentation algorithm features and the disease outcomes but no dependency with any other clinical information. Same trend has also been observed from the validation over an external dataset. Their survival prediction framework composed of two major unit (a) the U-Net Ronneberger et al. [2015] segmentation network and (b) the survival prediction model. The U-Net is trained with the dataset and physician contours and features are exported from the encoder end. Next, features are clustered in an unsupervised manner. Finally, a logistic regression model is trained for survival prediction. Motivated from this approach, we build a segmentation model to segment the LV first, then use the latent features of the encoder latent space for disease classification.

2 Methodology

We propose a deep learning framework that learns the segmentation related features from LV. Using the learned features, we trained several machine learning and deep learning models for disease prediction. We further investigated on the learned features and propose a feature selection technique that can reduce the redundant features maintaining reasonable classification accuracy. The method workflow can be divided into 4 major steps: (1) LV segmentation, (2) Feature extraction, (3) Feature reduction, (4) Disease prediction. The overview of the framework is shown in fig 1.

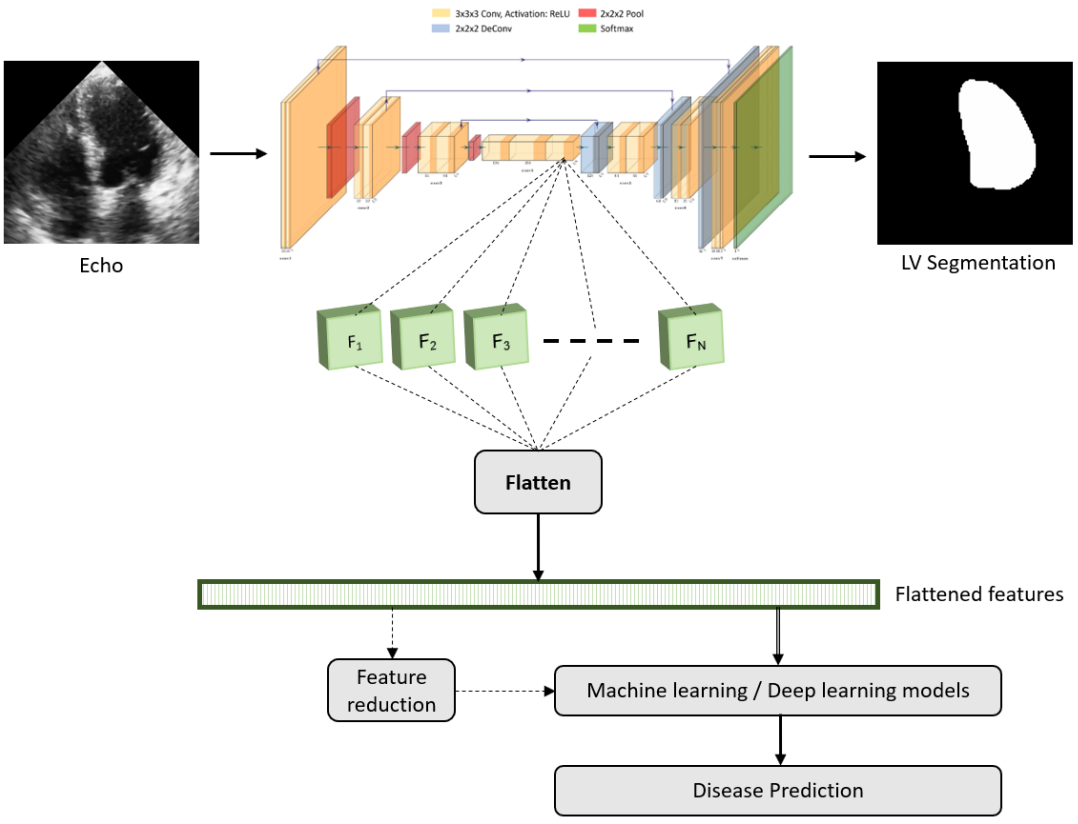

Figure 1: Method workflow diagram. A U-Net based architecture is used for LV segmentation from echo. Feature maps from the final convolutional layer of the encoder are used to train machine learning / deep learning models for disease prediction. Feature reduction block is further explained in 2

2.1 LV segmentation

We develop a U-Net based architecture to train and predict the error mask E_{pred}, which marks the erroneous regions of the segmentation mask I_{seg} in the input image I_{in}. Fig. **??** shows the proposed architecture. The network consists of 15 convolution layers with a kernel size of $3 \times 3 \times 3$ for each layer of convolution. The convolution stride is fixed to 1 voxel and padding is used to preserve the spatial resolution after each convolution. Max-pooling is performed over a $2 \times 2 \times 2$ voxel window, with a stride of 1.

Starting with 8 feature maps in the first convolution layer, the number of features is doubled after each two convolution layers up to the 6th convolution layer. The number of feature map is fixed to 32 for the 9th convolutional layer. After the 9th convolution layer, every step in the expansive path consists of an up-sampling of the feature map ("Up-convolution/De-convolution") that halves the number of feature channels, followed by concatenation with the correspondingly cropped feature map from the contracting path, and two $3 \times 3 \times 3$ convolutions. The final layer is a $1 \times 1 \times 1$ convolutional layer, with a soft-max activation. All hidden layers are equipped with a non-linear rectification (ReLU).

Let Seg-Net is the segmentation network that takes $I_{in} \in \mathbb{R}^{n \times n \times n}$ as the input echo to the segmentation network and produces $I_{seg} \in \mathbb{Z}^{n \times n \times n}$, the segmentation mask. We use Seg-Net for further analysis.

2.2 Feature extraction

The final convolution layer of the encoder of the Seg-Net has 32 kernels. Each of the kernels represents a feature activation map corresponding to I_{in}. Intuitively, the latent space in the encoder of the Seg-Net are enriched with features relating to shape, position, texture and topology of the corresponding LV. We take these activation maps of 32 kernels for each I_{in} in the training dataset and flattened them into 1-D vector. This feature vector has the dimension of 1×3584 that can be further used for training the machine learning / deep learning models for disease prediction.

2.3 Feature reduction

The latent space in the bottleneck of the Seg-Net has already encoded segmentation related features that are much smaller in dimension compared to a traditional DCNN classifier. We propose a feature selection technique based on the GradCAM Selvaraju et al. [2017]. Using ground truth and the probabilistic prediction of the LV, we back propagated the loss of Seg-Net to find out the weights of the kernels in the bottleneck. The feature reduction can be divided into two steps: (1) Obtaining kernel weights, (2) Reduction of kernels. The feature reduction workflow is shown in fig 2.

2.3.1 Obtaining kernel weights

Let, F^l, y^c, N represents the kernels of a convolution layer, class c related probabilistic output, number of kernels in the layer of interest, respectively. We can find the weights of the kernels through back propagation using the following equation,

$$\alpha_c^l = \frac{1}{N} \sum_i \sum_j \sum_k \frac{\delta y^c}{\delta F^l_{i,j,k}} \tag{1}$$

where α_c^l represents the kernel weight of F^l corresponding to y^c. Finally the GradCAM can be visualized by weighted combination of the feature maps followed by a *RelU* operation,

$$L_c = ReLU\left(\sum_l \alpha_c^l F^l\right) \tag{2}$$

where, L^c represents the GradCAM of layer L for class c. Fig 3 shows a set of test dataset and their corresponding GradCAM visualization.

2.3.2 Reduction of kernels

Each of the kernels has weight associated with the LV segmentation. We rank them based on their weights and pick 5 highest weighted kernels for each individual training dataset. Let S be the set of all such kernels. The rank of the kernels can be different for each individual cases due to different activation. We select the 3 most frequent kernels from the set S. Fig 4 shows the accumulated GradCAMS for 3 cases of a sample echo: (1) Accumulation of all kernels, (2) 5 highest weighted kernels and (3) 3 most frequent highest weighted kernels.

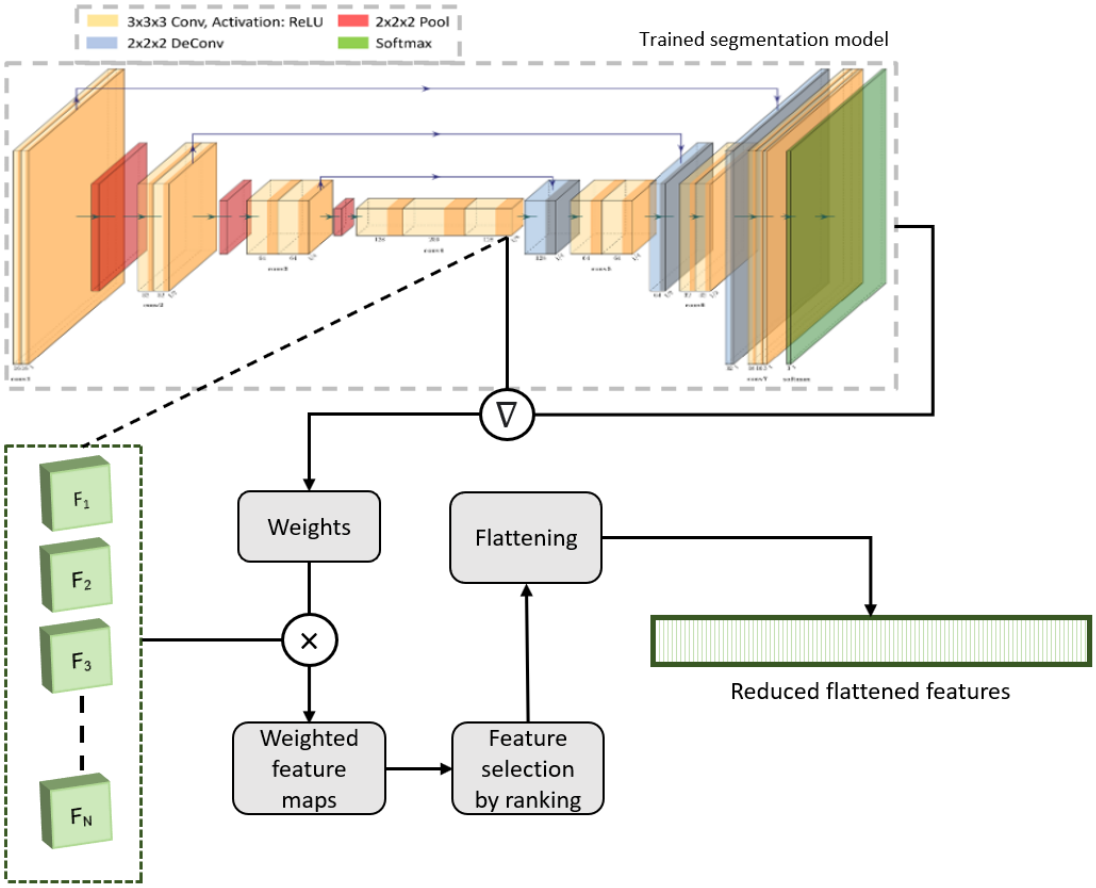

Figure 2: Feature reduction workflow. Bottleneck kernel weights are obtained through back-propagation. Then the Kernel features are selected based on their weighted ranks and frequency. Finally the reduced feature kernels are flattened and trained with machine learning / deep learning classifier.

The dimension of each kernel is $2 \times 7 \times 8$. So after flattening the reduced kernels, the dimension of the 1-D vector can be reduced to 560 and 336 for case (2) and (3) respectively. We use the reduced features in (3) to re-train the machine learning / deep learning models for disease prediction.

2.3.3 Disease prediction

Once the flatten feature matrix is obtained from the Encoder bottleneck of the U-NET, we see that number of feature dimensions is significantly larger than the number of observations for training. Support Vector Machines (SVMs) are highly effective in this case of supervised learning since SVM uses a subset of training points in the decision function also known as support vectors. Therefore, it is also memory efficient. When implementing SVMs for the binary classification of the high-dimensional feature space we propose, we have carefully chosen three kernels (Linear, RBF, and sigmoid) with an empirical choice of the values: "C" and "gamma". The choice of these hyper-parameters is crucial to get the balanced hyperplane and avoid over-fitting. We did not use polynomial kernel due to its inefficacy to generalize the decision boundary for very high dimensional datasets.

Implementing SVMs with different kernels, we see that SVMs with RBF kernel gives best classification accuracy among all the kernels. It is important to note that we used cross-validated grid-search over the parameter grid of Cs and Gammas. Next, we added Nu support to our choice of kernel to leverage the full potential of the SVMs and our hypothesis holds.

To achieve a) better prediction accuracy than any single contributing model by reducing variance; b) reduced spread in the predictions by the model, we used ensemble learning. An ensemble learning refers to machine learning model that combines the predictions from two or more models. There are three types of ensemble learning: boosting, bagging, and

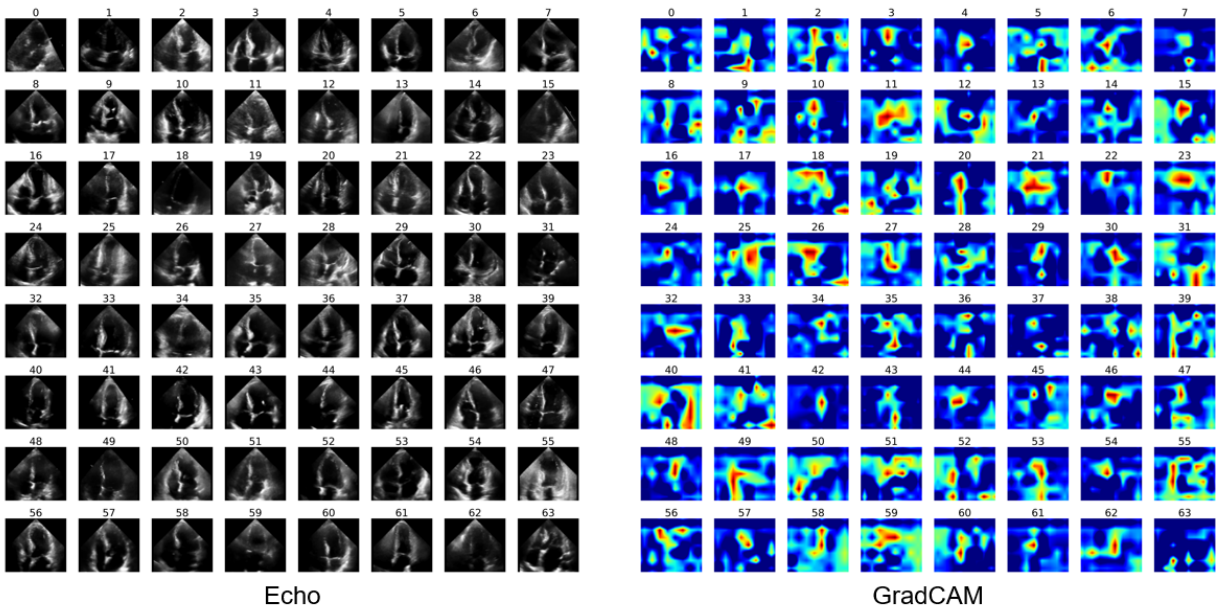

Figure 3: Example of a echo dataset with its corresponding GradCAM visualization.

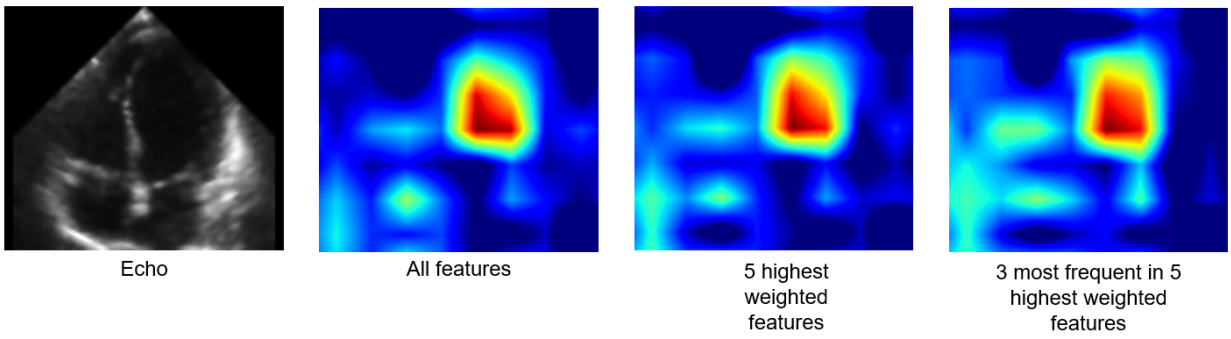

Figure 4: Example of a sample echo frame with its corresponding GradCAM representations for different number of feature kernels.

stacking. In boosting, the meta-estimator begins fitting the given estimator (RBF-SVC with for our case) to the training dataset and then fits additional copies on the same dataset but where the weights of incorrectly classified instances are adjusted such that subsequent classifiers focus more on the difficult cases. Unlike boosting, bagging fits the copies base classifiers on random subsets of the training dataset and then aggregate their individual predictions (either by voting or by averaging) to form a final prediction. Bagging emphasizes more on reducing the variance of the base estimator while boosting adjusts the weight of an observation based on the classification of the previous iteration.

During machine learning evaluation of our latent space, we used 3-fold cross validation to make sure the classification accuracy is stable. While implementing SVMs with different kernels, the accuracy for different estimators varied by no more than 3%, which means any of the folds might be estimated well by one of the kernels. Therefore, to accommodate the power of all kernels, we have used stack of SVMs with all three kernels with a final logistic regression-based classifier. This way, we were able to use the strength of each individual estimator by using their output to as input to a final estimator.

Table-1 reports all the estimators and their ensembles used in this binary classification with the empirical choice of parameters for reproducible research.

We further trained an Multi-Layer-Perceptron (MLP) for disease prediction. This neural network consist of 3 dense layers consisting 64, 256 and 512 neurons consecutively. Each of the dense layers are equipped with a dropout layer

Table 1: Estimators and their ensembles used in this binary classification

Estimator	**Hyper-parameters**
Linear-SVC	kernel : "linear", C : 1.0, gamma : "auto"
Sigmoid-SVC	kernel : "sigmoid", C : 1.0, gamma : "auto"
RBF-SVC	kernel : "rbf", C : 1.0, gamma : "auto"
Nu-SVC	nu : 0.7, kernel : "rbf", gamma : "auto"
Ensemble-Boosting	base_estimator : SVCrbf, n_estimators : 100, learning_rate : 0.0001, algorithm : "SAMME.R"
Ensemble-Bagging	base_estimator : SVCrbf, n_estimators : 150, bootstrap : False, bootstrap_features : True
Ensemble-Stacking	estimators:SVC with all three kernels, final_estimator:LogisticRegression (penalty:"l1", solver:"liblinear", C:1.0)

to avoid overfitting. The output layer has 2 neuron for 2 disease class prediction with soft-max activation. Binary cross-entropy is used as loss function.

3 Experimental Setup

We evaluated our method in an echo dataset obtained from Uiowa hospital. The dataset consist of 140 TTS, 160 STEMI and 150 Control cases. We only used TTS and STEMI cases for training and testing in method development. We did 3-fold cross validation for the entire dataset. The proposed segmentation method was implemented using Tensorflow. The network was optimized with the Adam optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$). The learning rate was initialized as $1e-6$ with an exponential decay rate of 0.05 after each epoch. The dataset was normalized to a zero mean and a unit standard variance. To reduce overfitting for the intensity variation caused by the MR scans for different subjects, we augmented every 3D MR image by adding a number uniformly sampled in between –0.1 and 0.1 to each image voxel (note that the intensity of each voxel was normalized) for randomly augmenting the image intensities.

4 Results

Fig 5 shows the 3-fold cross-validation classifications accuracy obtained from all the estimators and their ensemble methods. Accuracies reported by each of the base SVM estimators support our rationale described above. Among all the ensembles, we see that bagging of SVM with RBF kernel performs best. It is reasonable since bagging promotes generalizability by reducing spread of the predictors. Finally, we found neural net to perform best due to its high-dimensional flexibility to guarantee global minima.

Fig 6 shows 3-fold cross-validation classifications accuracy of the estimators when feature dimension is reduced by 10-fold. Certainly, this is a optimal trade-off between generalizability, computational complexity and prediction accuracy.

5 Discussion

Fig 5 shows that using the latent space of a segmentation network can achieve better classification accuracy than a direct DCNN classifier. The advantage of using latent feature space is that the features are more robust to noises and artifacts. Moreover the pathological evidences suggest correlation of region specific features to the cardiac diseases. The segmentation network enforces feature learning related to the topology and the motion of LV which play significant role in TTS and STEMI differentiation.

We proposed a novel feature reduction technique for the segmentation latent space based on GradCAM. The features are obtained based on their weights and importance corresponding to disease class prediction. Fig. 6 shows that using only 3 most frequent features having larger weights in the set of 5 most largest weighted features of the training dataset produces comparable result to Fig 6. But this dramatically reduces the computational complexity and make the model more generalized.

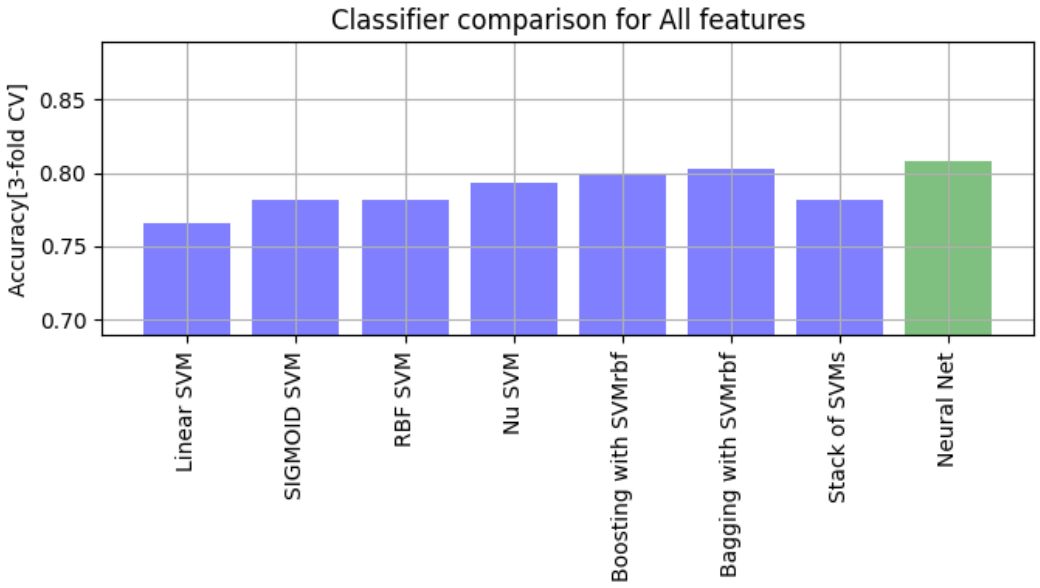

Figure 5: 3-fold cross validation accuracy from the machine learning / deep learning estimators considering all the feature kernels.

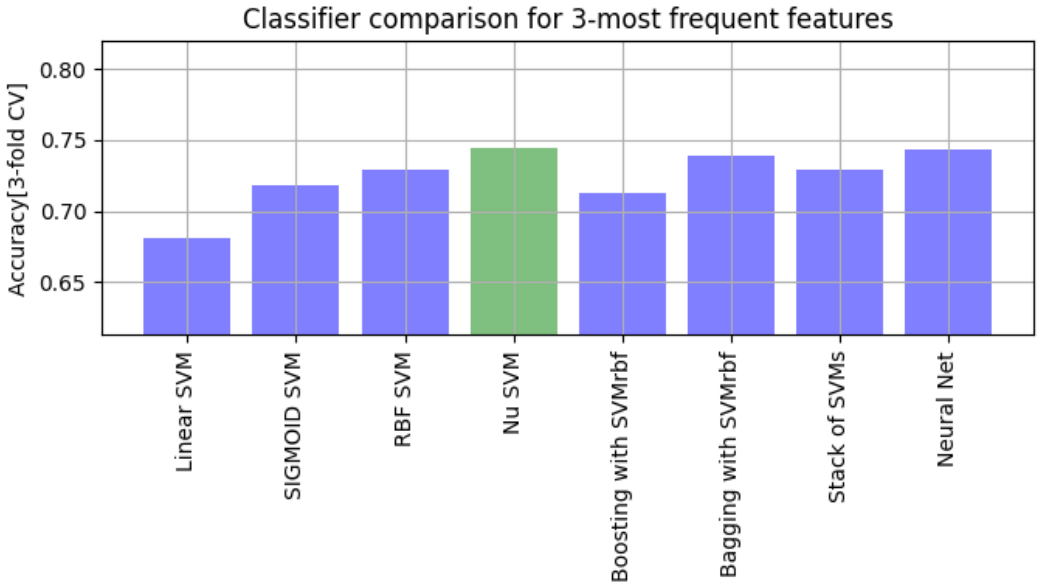

Figure 6: 3-fold cross validation accuracy from the machine learning / deep learning estimators considering the 3 most highest weighted feature kernels.

6 Conclusion

We obtained similar accuracy as the state-of-the-art approach using latent features from the segmentation network. Moreover, the obtained features are robust to noise and artifacts, as the segmentation network enforce learning from the object of interest only, specific to disease pathology. Our feature reduction technique shows a promising frame to reduce redundant features, thus reducing computational complexity and promoting network interpretability maintaining reasonable accuracy. Finally, our approach is a significant step towards disease prognostic prediction pipeline as it supports robust feature selection relating to organs of interest.

References

Fahim Zaman, Rakesh Ponnapureddy, Yi Grace Wang, Amanda Chang, Linda M Cadaret, Ahmed Abdelhamid, Shubha D Roy, Majesh Makan, Ruihai Zhou, Manju B Jayanna, et al. Spatio-temporal hybrid neural networks reduce erroneous human "judgement calls" in the diagnosis of takotsubo syndrome. *EClinicalMedicine*, 40:101115, 2021.

Fabian Laumer, Davide Di Vece, Victoria L Cammann, Michael Würdinger, Vanya Petkova, Maximilian Schönberger, Alexander Schönberger, Julien C Mercier, David Niederseer, Burkhardt Seifert, et al. Assessment of artificial intelligence in echocardiography diagnostics in differentiating takotsubo syndrome from myocardial infarction. *JAMA cardiology*, 7(5):494–503, 2022.

Stephen Baek, Yusen He, Bryan G Allen, John M Buatti, Brian J Smith, Ling Tong, Zhiyu Sun, Jia Wu, Maximilian Diehn, Billy W Loo, et al. Deep segmentation networks predict survival of non-small cell lung cancer. *Scientific reports*, 9(1):1–10, 2019.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.