2021 43rd Annual International Conference of the
IEEE Engineering in Medicine & Biology Society (EMBC)
Oct 31 - Nov 4, 2021. Virtual Conference

Semi-Supervised Analysis of the Electrocardiogram Using Deep Generative Models

Søren M. Rasmussen[1], Malte E. K. Jensen[2], Christian S. Meyhoff[4,5,6]*,
Eske K. Aasvang[3,5]*, Helge B. D. Sørensen[1]**Senior Member*, IEEE

***Abstract*— Deep learning has gained increased impact on medical classification problems in recent years, with models being trained to high performance. However neural networks require large amounts of labeled data, which on medical data can be expensive and cumbersome to obtain. We propose a semi-supervised setup using an unsupervised variational autoencoder combined with a supervised classifier to distinguish between atrial fibrillation and non-atrial fibrillation using ECG records from the MIT-BIH Atrial Fibrillation Database. The proposed model was compared to a fully-supervised convolutional neural network at different proportions of labeled and unlabeled data (1%-50% labeled and the remaining unlabeled). The results demonstrate that the semi-supervised approach was superior to the fully-supervised, from using as little as 5% (5,594 samples) labeled data with an accuracy of 98.7%. The work provides proof of concept and demonstrates that the proposed semi-supervised setup can train high accuracy models at low amounts of labeled data.**

I. INTRODUCTION

Deep learning has in recent years had an increasing impact in medical research, where models can be trained to very high performance in a growing number of medical fields. One of these fields are cardiac arrhythmia. However, the increase in performance has mainly been driven by supervised methods, requiring larger datasets that have been very expensive obtain.

Medical data are quite expensive to properly label, as it often requires specially trained and experienced medical staff. Hence, while vast amounts of medical data exist, only a small amount of it has high quality labels. Further, the price of labeling often means that these datasets are not publicly available. One solution to this, is to use semi-supervised learning, where an unsupervised model is jointly trained on large amounts of unlabeled data with a supervised model that is trained on a smaller amount of labeled data.

* shared last authorship
Corresponding author Søren M. Rasmussen (email: smora@dtu.dk)
First authorship is shared between Søren M. Rasmussen and Malte E. K. Jensen
[1] Department of Health Technology, Technical University of Denmark, Kongens Lyngby, Denmark
[2] Cluster for Molecular imaging, University of Copenhagen, Copenhagen, Denmark
[3] Department of Anaesthesiology, Centre for Cancer and Organ Dysfunction, Rigshospitalet, University of Copenhagen, Copenhagen, Denmark
[4] Department of Anaesthesia and Intensive Care, Bispebjerg and Frederiksberg Hospital, University of Copenhagen, Copenhagen, Denmark
[5] Copenhagen Center for Translational Research, Copenhagen University Hospital, Bispebjerg and Frederiksberg, Copenhagen, Denmark
[5] Department of Clinical Medicine, University of Copenhagen, Copenhagen, Denmark

As a case study, this paper focuses on detection of atrial fibrillation (AF) from electrocardiogram (ECG) signals. According to the National Health Service (NHS) AF is the most common heart rhythm disturbance affecting more than 1 million people in the United Kingdom alone [1]. AF is classified as a tachyarrhythmia, where the electrical impulse is not initiated in the sinus node, but instead in fibrillatory waves in the atrias. AF is characterized as an irregular rhythm with loss of the P-waves in the ECG signal.

Detection of AF is a topic that is well investigated with a lot of the methods focusing on the irregularity of the R-R intervals (RRI) as their main feature [2], [3]. Using only the RRI Hong-Wei et al. achieved 97.8% and 99.0% for sensitivity and specificity respectively [2], and Faust et al. achieved 99.9% and 99.61% [3]. However, the use of the RRI tend to result in a high degree of false positives when the R-peaks are falsely detected. When using common R-peak detection algorithms such as the Pan-Tompkins [4], this is common in case of noise [5]. He et al. proposed a method using the ECG as input and achieved a sensitivity of 99.4% and a specificity of 98.9% [6].

Other attempts on using semi-supervised learning for arrhythmia in ECG signals have been made. Zai et al. have made an ectopic beat classifier, but this however needs retraining for each new patient. Costa et al. have used an VAE to classify AF on simulated and real data, but using intracardiac recordings from pacemaker systems.

For the semi-supervised model, we propose the use of a variational autoencoder (VAE), which is an unsupervised deep generative model (DGM) originally proposed by Kingma et al. [9]. The VAE is a network in which a high dimensional input is mapped into a low dimensional latent space, from which a high dimensional reconstruction is created, hereby forcing the network to compute features that describe the input signal. Previously the use of a VAE as a semi-supervised auxiliary deep generative model was demonstrated by Maaløe et al. [10].

We hypothesize that by training a deep neural classification model in a semi-supervised approach, it will be possible to obtain performance on par with the state of the art by only using a small proportion of labeled data.

II. METHODS

A. Data

The data used in this project came from the MIT-BIH Atrial Fibrillation database (AFDB) [11], [12]. The AFDB consists of 23 ECG records, each with two leads and of 10 hours

length. The records have been digitized using a sampling frequency of 250 Hz and a 12-bit resolution in the $\pm 10mV$ range. Unaudited annotations of the QRS complexes are available along with manual annotations of the following subcategories: AF, Atrial Flutter, AV-Junctional rhythm and Sinus Rhythm (SR).

B. Preprocessing of Data

Each ECG record was split into 10 seconds non-overlapping segments, to avoid parts of the same segment being present in both the labeled and unlabeled data set. The label was given based on the annotation files available with the data and was divided into AF vs. Non-AF. In situations where multiple labels were present in the same segment, the label present for the majority of the segment was used for the entire segment. For both the training and test set, the data was balanced by down-sampling of the majority class. The dataset was split into a training set containing 90% of the segments and a test set containing the remaining 10%.

To remove the DC-offset and any baseline wandering before normalization, a high-pass filter with cut off frequency of $0.5Hz$ and a filter order of 5 was utilized. All segments were down sampled to $100Hz$

C. Variational Autoencoder

The VAE is an unsupervised generative model, that consists of two neural networks: An inference model, the *encoder* and a generative model, the *decoder*. The encoder maps the input sample into a set of lower dimensional latent variables, which the decoder maps into a reconstruction of the input sample. The VAE builds upon probability theory and Bayes' rule. In the VAE the inference model is defined as $q_\phi(z|x)$ and the generative model as $p_\theta(x|z)$ [9]. By including the label variable, y, into the model, a semi-supervised generative probabilistic model can be achieved [10]. In this model the inference model, *Q*, is defined as $q_\phi(z|x,y)q_\phi(y|x)$, with each term defined as:

$$q_\phi(z|x,y) = \mathcal{N}\left(z|\mu_\phi(x,y), \text{diag}\left(\sigma^2_\phi(x,y)\right)\right) \quad , \tag{1}$$

$$q_\phi(y|x) = \text{Bernoulli}\left(y|\pi_\phi(x)\right) \quad , \tag{2}$$

and the generative model, *P*, is defined as $p(z)p_\theta(x|z,y)$, with each term defined as:

$$p(z) = \mathcal{N}(z|0,\mathrm{I}) \quad , \tag{3}$$

$$p_\theta(x|z,y) = f(x;z,y,\theta) \quad , \tag{4}$$

where q_ϕ and p_θ are neural networks with parameters ϕ and θ, respectively. The inference and generative model is shown in Figure 1.

The Gaussian distribution $q_\phi(z|x,y)$ is achieved by splitting the last layer of the model into two channels representing the mean, μ_ϕ, and the log variance, $\log\sigma^2_\phi$ of the distributions, from which z is sampled using the reparameterization trick[13].

The reconstruction loss $p(x|z,y)$ is defined as a Gaussian distribution with μ_θ being the reconstruction and $\sigma_\theta = 2$.

The objective of optimizing the parameters, θ and ϕ, is to maximize the log-likelihood $\log p(x)$. This is achieved by using Jensen' inequality to obtain the evidence lower bound function, which can be optimized. For the unlabeled case, the lower bound is given as

$$\begin{aligned}\log p(x) &= \log \int_z \sum_y p(x,y,z)dz \\ &\geq E_{q_\phi(z,y|x)}\left[\log \frac{p_\theta(x,y,z)}{q_\phi(z,y|x)}\right] \equiv -\mathcal{U}(x) \quad ,\end{aligned} \tag{5}$$

and for the labeled case, the lower bound is defined as

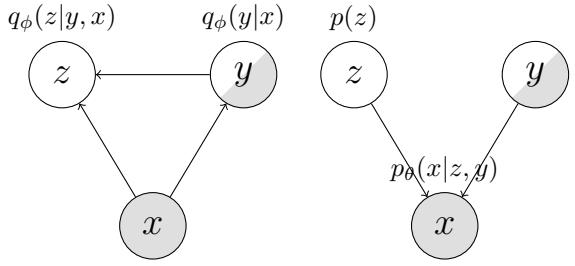
$$\begin{aligned}\log p(x,y) &= \log \int_z \sum_y p(x,y,z)dz \\ &\geq E_{q_\phi(z|x,y)}\left[\log \frac{p_\theta(x,y,z)}{q_\phi(z|x,y)}\right] \equiv -\mathcal{L}(x,y) \quad .\end{aligned} \tag{6}$$

In the lower bounds the contribution of z and y in the unlabeled case and z in the labeled case is marginalized out. For the unlabeled case, y is treated as latent variable and is sampled by summing over the two classes. For z, the integral is approximated by sampling from the Gaussian distribution in the latent space. In the case of labeled data, optimization for the labels, y, is done using binary cross-entropy.

D. Loss Functions and Warm Up

Besides the lower bounds defined in (5) and (6), an extra loss was introduced, consisting of the absolute difference between the standard deviations of the input and reconstruction. This was introduced to help the decoder to make better reconstructions. For the classifier, binary cross-entropy loss was used.

To further help the training of the DGM two warm ups were introduced, defined as delay and a linear ramp up to a maximum value. One for the KL divergence, with a 25-epoch delay, a max weight of 0.1 at epoch 100, and a second for the classification loss, without delay and a max weight of 0.5 at epoch 40. These were introduced to avoid restraining the generative part of the network too much in the beginning, before pushing towards classification and a standard normal distribution for z.



(a) Inference Model, *Q* (a) Generative Model, *P*

Fig. 1. Diagram of (a) the inference model and (b) the generative model of the proposed network. The grey color of the nodes denotes known data, and the partly colored node labeled y emphasized the semi-supervised aspect of the model.

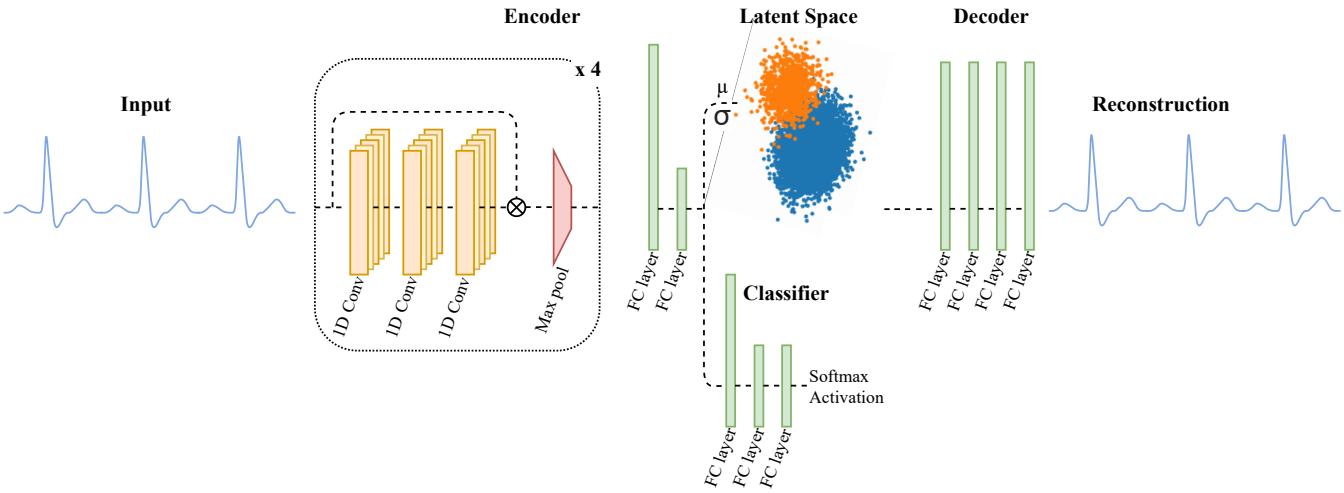


Fig. 2. Diagram of the proposed deep generative model. *1D Conv:* 1-dimensional convolutional layer. *FC layer:* Fully connected layer.

E. Model Structure

The DGM can be divided into three parts, the encoder, the classifier and the decoder. The encoder was built with a residual network (ResNet) architecture consisting of four blocks each containing three convolutional layers and a residual connection. ResNet has shown superiority in other image classification tasks, when compared to classic convolutional networks [14]. In order to increase the receptive field of the network, dilation of 2, 4 and 8 was applied to the three layers within each block respectively. Max-pooling was done in the end of each block using a kernel size of 3 and a stride of 3, thus decreasing the signal size by a factor 3 per block. The kernel size and stride were 3 and 1 respectively for all convolutional layers, and the number of output channels were fixed per block to 32, 32, 64, 64 for the four blocks respectively. Two fully connected layers were applied to the end of the blocks with a size of 1,000 and 500. The decoder and the classifier were constructed as simple fully connected neural networks. It consisted of an input layer, four hidden layers each with 4,096 nodes and an output layer. The classifier consisted of three layers with 500, 200 and 200 nodes respectively and an binary softmax function as output. All layers except for output layers used Rectified Linear Unit as activation function and had batch normalization and dropout ($p = 0.3$). A diagram of the model is shown in Figure 2.

F. Evaluation of the Proposed Model

In order to demonstrate the potential of using the semi-supervised approach, the proposed DGM was tested against a conventional CNN, identical to the encoder + classifier of the DGM. The setup was constructed using different proportions of unlabeled and labeled data, where the labeled data were used to train both the DGM and the CNN, and the unlabeled part of the data were used only to train the DGM with the unupervised loss. In this way a data "titration curve" setup was obtained mimicking scenarios where different amounts of labeled data could be obtained data. The models were trained in setups using 1%, 5%, 10% and 50% of the data as labeled and the remaining as unlabeled. It was ensured that for each setup, the data in the training and test set were the same for both the DGM and CNN. Furthermore the random seed was fixed such that the runs were kept as similar as possible.

A total of 111,894 segments were available in the training set after balancing the classes. The test set consisted of 12,434 segments that were also balanced. Each training phase of the DGM consisted of 50 epochs, where labeled data were cycled to correspond with the amount of unlabeled data. As the amount of data per epoch was less when training the CNN, and hence less updates if it was only permitted to train for 50 epochs, these were allowed to train until convergence to give more fair conditions.

III. RESULTS

The results of the training of the DGM and the CNN using different amounts of labeled data are shown in Table I. The highest absolute performance was obtained by the DGM in the semi-supervised approach using 50% of the data labeled, whereas the largest difference between the CNN and DGM was obtained when using 1% of the data as labeled.

An input segment and corresponding reconstruction of chosen samples are shown in Figure 3, and the distribution of the samples for the test set in the latent space is shown in Figure 4

TABLE I

TITRATION CURVE OF THE PERFORMANCE ACHIEVED BY THE PROPOSED DGM AND THE REFERENCE CNN RESPECTIVELY AT DIFFERENT AMOUNTS OF LABELED DATA.

No. of labeled data	Accuracy					
	DGM			CNN		
	Acc	Sen	Spe	Acc	Sen	Spe
1% (1,118)	94.0%	98.8%	89.3%	65.8%	93.6%	38.0%
5% (5,594)	98.7%	98.5%	98.9%	95.3%	98.5%	92.1%
10% (11,190)	98.7%	98.9%	98.5%	97.7%	96.2%	99.1%
50% (55,948)	**98.8%**	98.9%	98.8%	98.2%	97.5%	99.0%

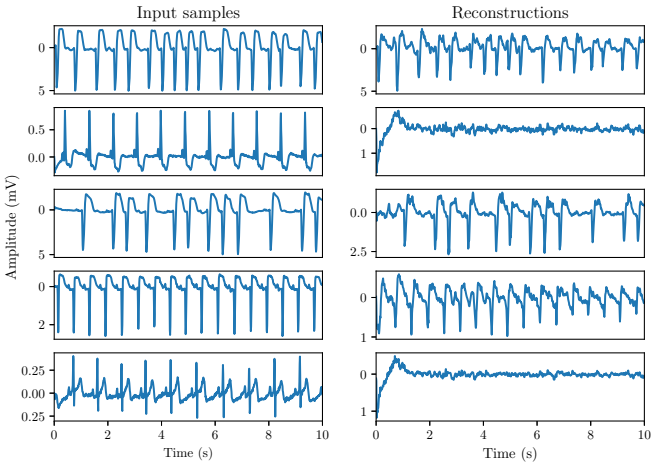


Fig. 3. Examples of input images and the corresponding reconstructions achieved by passing samples through the DGM trained on 1% labeled data.

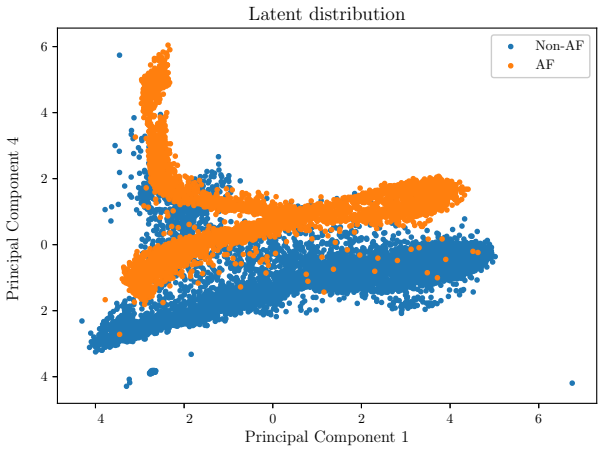


Fig. 4. Scatter plot of the latent space of the inputs from the test set transformed into 2 dimensions using principal component analysis. The DGM used to obtain the data points was trained on 10% labeled data.

IV. DISCUSSION

The purpose of this study was to provide a proof of concept that by using a semi-supervised setup when training a deep neural network, unlabelled data could increase performance above only supervised. The results in the titration curve in Table I show that the proposed semi-supervised approach achieves higher performance in all test cases, with the most prominent difference in the cases with lower amount of labeled data. Comparing the results at different amounts of labeled data even shows, that the semi-supervised approach with 5% labeled data was superior to the highest obtained performance by the fully-supervised approach in the 50% labeled data. Even in the case of 1% labeled data, the DGM achieved an accuracy of 94.0%, which despite not being directly on par with the state of the art or the highest achieved performance in this setup, was still impressive with just above one thousand labeled data points.

The use of the DGM in combination with a conventional fully-supervised classifier is based on the idea, that the loss from the reconstructions will teach the network the structure of the data. In order to achieve that, the reconstructions should differ and optimally be like the input signals. The reconstructions shown in Figure 3 show that even though it in some cases did not capture the QRS complexes, it obtained a good reconstruction of other samples. This shows that the network was capable of learning from the signals, also helping the classifier to learn.

Our work showed very promising results for DGMs in semi-supersived learning, despite using a datasebase that was not ideal for deep learning due to it's small size. Hence, future research should be aimed at larger datasets to further explore the potential of DGMs for semi-supervised learning in training high-performing models in ECG signals.

V. CONCLUSIONS

We began this paper by hypothesizing that by using DGM to create a semi-supervised setting for training a deep neural network, it would be possible to achieve close to state of the art performance on only a small amount of labeled data. The work presented here thus demonstrates the potential of using DGM for semi-supervised learning, and that in a setting where only a small amount of labeled data is available, information can be extracted effectively from the unlabeled data and increase classification performance.

REFERENCES

[1] NHS [Online], https://www.nhs.uk/conditions/atrial-fibrillation/

[2] L. Hong-Wei, S. Ying, L. Min, L. Pi-Ding, Z. Zheng, "A probability density function method for detecting atrial fibrillation using R–R intervals", Med. Eng. Phys., vol. 31, no. 1, Jan. 2009

[3] O. Faust, A. Shenfield, M. Kareem, T. R. San, H. Fujita, U. R. Acharya, "Automated detection of atrial fibrillation using long short-term memory network with RR interval signals", Comput. Biol. Med., vol. 102, Nov. 2018

[4] J. Pan, W. J. Tompkins, "A Real-Time QRS Detection Algorithm'", IEEE Trans. Biomed. Eng., vol. 32, Mar. 1985

[5] Z. F. M. Apandi, R. Ikeura, S. Hayakawa, S. Tsutsumi, "An Analysis of the Effects of Noisy Electrocardiogram Signal on Heartbeat Detection Performance", Bioengineering, vol. 7, no 2, Jun. 2020

[6] R. He, K. Wang, N. Zhao, Y. Liu, Y. Yuan, Q. Li, H. Zhang, "Automatic Detection of Atrial Fibrillation Based on Continuous Wavelet Transform and 2D Convolutional Neural Networks", Front. Physiol., vol. 9, Aug. 2018

[7] X. Zhai, Z. Zhou, C. Tin, "Semi-supervised learning for ECG classification without patient-specific labeled data", Expert Syst Appl, Vol 158, Nov. 2020,

[8] N. Costa, L. Sánchez and I. Couso, "Semi-Supervised Recurrent Variational Autoencoder Approach for Visual Diagnosis of Atrial Fibrillation," IEEE Access, vol. 9, Mar. 2021,

[9] D. P. Kingma and M. Wellin, "Auto-Encoding Variational Bayes", arXiv, May 2014

[10] L. Maaløe, C. K. Sønderby, S. K. Sønderby, O. Winther, "Auxillary Deep Generative Models", arXiv, Jun. 2016

[11] G. B. Moody and R. G. Mark, "A new method for detecting atrial fibrillation using R-R intervals", Comput. Cardiol, vol. 10, 1983

[12] A. Goldberger, L. Amaral, L.Glass, J. Hausdorff, P. Ivanov, R. Mark, J. E. Mietus, J. E. Moody, C. K. Peng and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals", Circulation, vol. 101, no. 23, Jun 2000

[13] C. Doersch, "Tutorial on Variational Autoencoders", arXiv, Jan. 2021

[14] K. He, X. Zhang, S. Ren, J. Sun, "Deep Residual Learning for Image Recognition", arXiv, Dec. 2015