12-lead ECG classification using Explainable Neural Networks and Ensemble models

Bjørn-Jostein Singstad[1]
[1] **Department of Physics, University of Oslo, Oslo, Norway**

E-mail: b.j.singstad@fys.uio.no

Abstract.

Existing, commercially available ECG algorithms are mainly rule-based and have limitations in their accuracy. Machine learning, which have shown great performance in many fields over the last years, can possibly outperform the existing ECG-algorithms. This study builds on the Physionet/Computing in Cardiology challenge 2020 which aimed to classify multiple diagnoses based on 43101 12-lead ECGs. The models presented are convolutional neural networks and ensemble models based on clustering and random forests. These models are complex and often seen as black boxes in terms of explainability. This study addresses this problem by showning how local interpretable model-agnostic explanations (LIME) can be used to explain the predictions of such models.

The best Ensemble model, utilizing features from all 12 leads, outperformed the convolutional neural networks in this study, with a average cross-validated Physionet/Computing in Cardiology Challenge score of 0.512 ± 0.006. This score are only 0.021 behind the cross-validated score, on the same development set, reported by the winner of the Physionet/Computing in Cardiology Challenge 2020.

1. Introduction

Cardiovascular diseases is one of the leading causes of death in the world. Numbers from WHO estimates that 17.9 million people died from cardiovascular death (CVD) in 2016 which represented 31% of all global death that year (1). Early detection of patient with risk of CVD could potentially decrease the amount of CVD. Electrocardiography is a method with a huge potential for detecting patients with risk of CVD. The electrocardiograph is non-invasive and relativly easy to use, compared to methods like echocardiogram and MRI, which makes it an convenient diagnostic tool. As an example of how widely the electrcardigraph is used, National Ambulatory Medical Care reported that 40 millions electrocardiograms (ECG) were recorded in USA in 2015 (2).

An electrocardiograph measures the electrical activity of the heart from electrodes placed on the surface of the upper body. The result of such a measurement is an ECG. The ECG is a graphical representation of the measured electrical activity of the heart with respect to time. One of the challenges is that the ECG can be difficult to interpret correctly. The interpretation can be time consuming and require a high degree of expertise (3).

Many of the modern and clinically used electrocardiographs today are equipped with a built-in interpretation program. The interpretation program analyzes the ECG and prints interpretive texts that may indicate different diseases. Studies show that there are some limitations to the automatic interpretation algorithms (4; 5). The errors, caused by the automatic interpretation algorithms, means that doctors or cardiologists has to read over the ECGs to ensure they are correct.

The hypothesis in this study is that machine learning can improve today's interpretive algorithms. Eight machine learning models from a previous study (6) will be evaluated and compared with two new machine learning models, were one of them will utilize 12 lead and the other will utilize only 2 leads.

A considerable amount of literature has been published on heartbeat classification (7), single (8) and even 2-lead classifiaction (9) over the last ten years . In most recent years there have been an increasing focus on 12-lead ECG classificaion and some recent studies has shown that machine learning is feasible (10; 11; 12; 13). On the other hand, the dataset used has either been small and homogeneous (14) or not accessible to everyone. In this study, a large, open dataset from several sources and a large variation in different diagnoses will be examined and used as development set for training machine learning models (15). This dataset was used in a challenged held by PhysioNet (16) and Computing in Cardiology (CinC) in 2020 were 217 teams submitted 1395 algorithms during the challenge (15). A training set and a test set were provided and the team who got the best score on the test set won the competition. The best team called themselves *prna* and they achieved a PhysioNet/CinC Challenge score (15) of 0.533 on the test set and a cross-validated score of 0.533 ± 0.046 (mean and standard deviation).

It is already stated that PhysioNet/CinC Challenge 2021 will utilize the same dataset, but this time investigate both 12-lead and 2-lead ECG in a challenge called

"will 2 to?". One of the objective of this study is to prepare an initial submission to the PhysioNet/CinC Challenge 2021 which will go live in the end of December 2020.

In addition, this study will demonstrate how to get an explainable prediction from the machine learning models developed in this study. Explainability of machine learning models is a new field in artificial intelligence (AI) and is called explainable AI. Explainable predictions are very important in medical diagnostics. As an example, cardiologists and doctors can use their knowledge to see if the parameters used for the prediction, by an ECG classification model, can be explained physiologically. This will probably lead to better trustworthiness for an ECG classification algorithm among cardiologists and doctors.

2. Methods

2.1. Data

The PhysioNet/CinC Challenge 2020 development set consisted of 43101 ECG-recordings. The datasets were sourced from six subset from four different sources:

- The first source is China Physiological Signal Challenge 2018 wich consists of two subsets: The original China Physiological Signal Challenge 2018 dataset (17) and a extra set called China Physiological Signal Challenge Extra.
- The second source is the Physikalisch-Technische Bundesanstalt (PTB) which consist of two subsets. The first one is the PTB Diagnostic (18) and the second subset is PTB-XL (19)
- The third source is the St. Petersburg Institute of Cardiological Technics (INCART) database (16)
- The fourth is the Georgia 12-Lead ECG Challenge Database which is new database and is still not described in any paper other than the PhysioNet/CinC Challenge 2020 paper (15).

A total of 111 different diagnoses were present in the total dataset. Each ECG-recording had at least one diagnosis, but some also had more than one diagnosis. The classification of such a data set is considered a multi-label, multi-class classification problem. The goal of PhysioNet / CinC Challenge 2020 was to classify 27 of the 111 diagnoses.

2.1.1. Splitting of data The data were splitted into training (90%) and validation (10%) data using 10-fold stratified cross-validation with random state = 42 (20). The stratification arranged the splitting such that the distribution of diagnoses was the same in both the train and validation data.

2.2. Preprocessing data

Initially the diagnosis were encoded with Systematized Nomenclature of Medicine Clinical Terms (SNOMED-CT). The SNOMED-CT codes were decoded into human

readable diagnosis and one-hot encoded into a 27-bit long array. Each of the bits in the array represented one of the 27 scored diagnosis in the PhysioNet/CinC Challenge 2020 (15). The 84 unscored diagnoses were overlooked and did not represent any change in the 27-bit long label array. The same labels were used in both the CNN-models and the ensemble models, the preprocessing and feature extraction from the ECG were done differently for the CNN model and for the ensemble model.

2.2.1. Preprocessing for Convolutional models

All ECG-recordings used by the CNN-models were padded or truncated to a signal length of 5000 samples. Padding and truncation were done by removing any parts longer than 5000 samples and adding a tail of $5000 - n$ zeros to any recording of length $n < 5000$. The rule-based model on the other hand, which was used in two of eight CNN-models, analyzed the ECG before padding or truncation to 5000 samples.

The 27 diagnoses/classes were not balanced and, to prevent the CNN-models to learn more from the diagnosis that occur more frequently in the dataset, a class weight were calculated. The weights were fed to the models during training and gave higher priority to ECGs with rare diagnosis than diagnosis that occur more frequently in the dataset.

2.2.2. Preprocessing for the ensemble models

All ECG recordings were fed into a ECG-featurizer function (21). The ECG-Featurizer analyzed the ECGs and extracted 112 features from the ECGs. All of the 112 features were used in the 12-lead classification while only 63 were used in the 2-lead classification. Only features that were extracted from lead II and V5 were used in the 2-lead model.

42720 of 43101 ECGs were successfully featurized på the ECG-featurizer. In addition, 146 ECG-recordings were removed due to missing values. This gave a total dataset of 42574 succesfully featurized ECGs to use by the ensemble model.

2.3. Model architectures

2.3.1. CNN architectures

The CNN architectures used in this study was the same as in (22). The models are listed in table 1. The new contribution in this study is that the 8 models were scored using cross-validation on the development data.

2.3.2. Ensemble model architecture

The architectures trained on the featurized ECG-data were build up using scikit-multilearn (23). Scikit-multilearn is a library for multi-label classification. The ensemble models built using Label Space Partitioning Classifier (24), Classifier Chain (25) and

Table 1: The eight CNN models developed in (22) and used in this study

Model
A) FCN
B) Encoder
C) FCN || age, gender
D) Encoder || age, gender
E) Encoder || FCN
F) Ecoder || FCN || age, gender
G) Encoder || FCN + rule-based model
H) Encoder || FCN || age, gender + rule-based model

random forests. Label Space Partitioning Classifier is a cluster algorithm were each cluster is a separate classification problem. The clusters were selected using a method called fixed label space clusterer. This method lets the developer define the clusters. The clusters in this study were created by iterating over all diagnosis in the training set and for the n-th diagnosis there was m_n diagnosis that co-existed with the n-th diagnosis across the data set. In total there was 27 clusters and the size of the clusters, $m_n + 1$, were different for each of the folds in the 10-folded cross-validation. This is illustrated in figure 1.

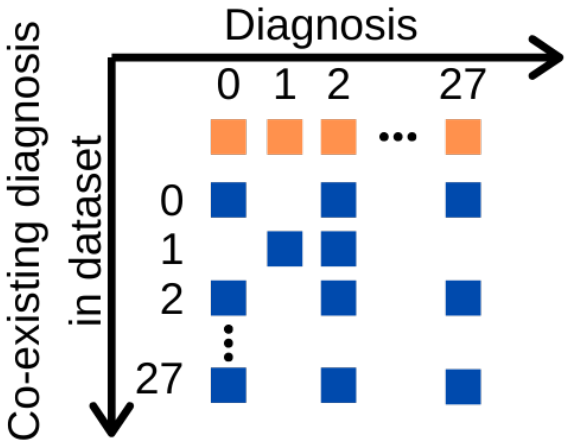


Figure 1: The figure shows how each of the 27 diagnosis co-existed with one or more of the other 26 diagnosis when looking across the whole development dataset. The 27 clusters that was used for the ensemble models used the same index as the co-existing diagnosis.

2.4. Threshold optimization

For the CNN-models a threshold for the 27 classes had to be set. New thresholds were set for each fold. A method called Nelder-Mead downhill simplex method (26; 27) was

used to optimize the thresholds individually with PhysioNet/CinC Challenge score as the optimization goal. This method can be computational heavy and therefore a subset of the training set were used to optimize the threshold. The subset were determined using a stratified 10-fold and then selecting the first validation fold as the threshold optimization set.

The Nelder-Mead downhill simplex method is used to find the local minimum of a function using the function itself and an initial guess of the optimal variable of the function. In the first fold of the 10-folded cross-validation the initial guess for the Nelder-Mead optimization algorithm was found by giving a 27-element long array values of 1 and multiplied it with a variable that was given values from 0 to 1, with a step size of 0.05. The next folds in In the 10-folded cross-validation used the threshold found by the Nelder-Mead in the previos fold as the initial guess.

The Ensemble models on the other hand did not need any threshold optimiztion as the output were binarized by the model.

2.5. Explanation models

To add explainability to the models used in this study a local interpretable model-agnostic explanation (LIME) was used (**?**). LIME explains the feature importance locally for a given prediction. LIME uses a linear model to explain the prediction of the the mode complex CNNs and ensemble models used in this study. As a proof-of-concept for such explaination models the 12-lead ensemble model and the CNN encoder model were selected to be explained. The ensemble model were explained using a tabular explainer-function.

To explain the CNN model a explainer called recurrent explainer was used. The Encoder model were simplified by swaping the last layer in the Encoder-model with a softmax-layer with two nodes and the dataset were modified such that Normal Sinus Rhythm were equal to Normal class and all other diagnosis were equal to Abnormal class.

3. Results

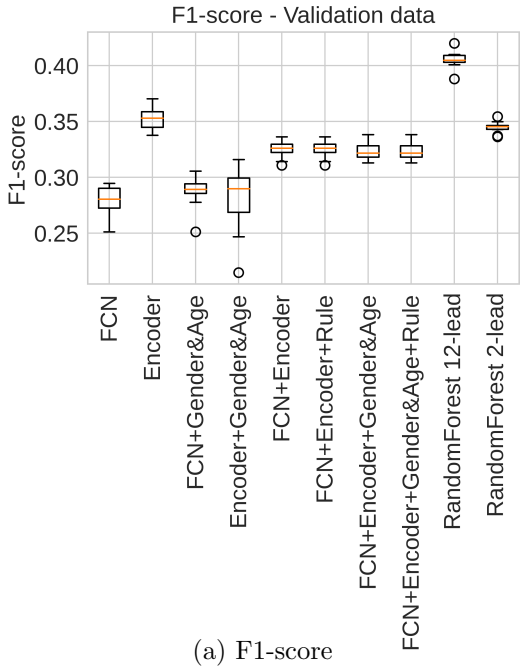
3.1. Cross-validated results

The CNN models were trained using ADAM-optimizer, a batchsize 30 and the Area Under the Curve (AUC)-score, on the validation set, was used to reduce learning rate during training. The learning rate was initially at 0.001 for all models and decreased by a factor of 10, each epoch that the AUC score did not improve. Early stopping used to stop training when AUC score, on the validation data, did not improve over two successive epochs.

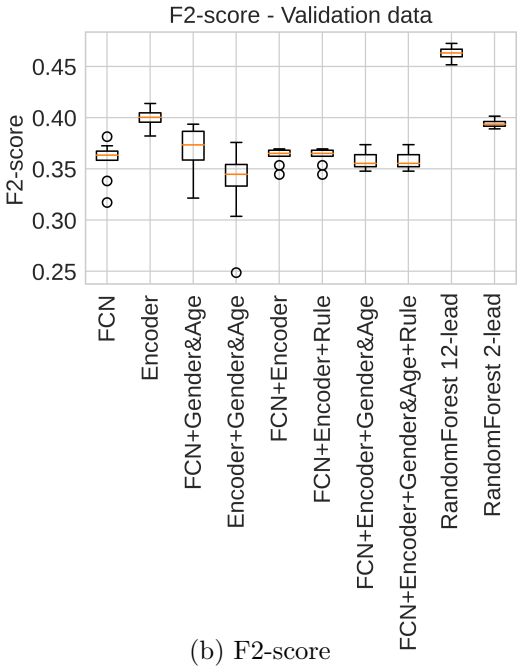
The ensemble models were trained using n_estimators = 3 for the random forest classifier and the input features where scaled using a Standard Scaler (20).

All models were scored using F1, F2, G2 and PhysioNet CinC Challenge score for each of the 10-folds ‡. The results in figure 2 shows that the random forest-based model, using features from all 12 leads, outperformed the other model on all metrics used in this study.

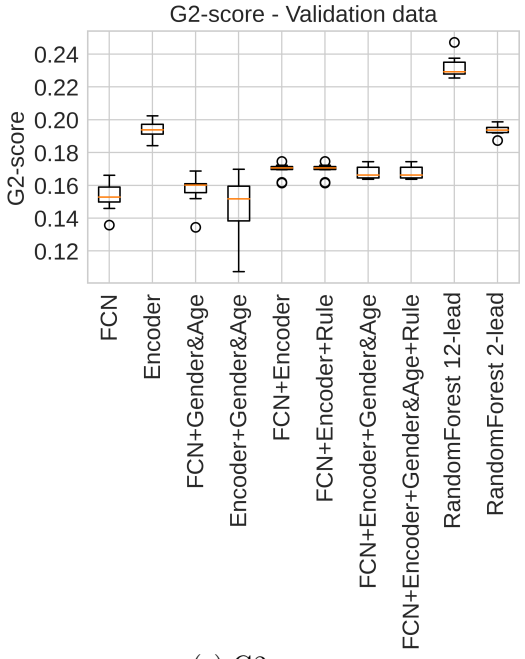
‡ All codes and models are available here: https://github.com/Bsingstad/FYS-STK4155-oblig3



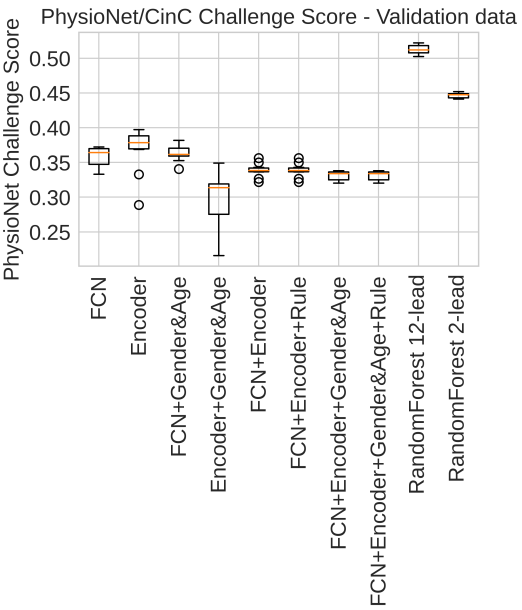
(a) F1-score



(b) F2-score



(c) G2-score



(d) PhysioNet/CinC Challenge-score

Figure 2: The figure shows 10-fold cross-validated scores achieved by ten different models. The upper left show F1-score, the upper right show F2-score, the lower left show G2-score and the lower right show PhysioNet/CinC Challenge score. The PhysioNet/CinC Challenge score are described in (15)

3.2. Explainability results

The tabular explainer that was applied on the ensemble model was trained on 5000 ECGs from the training data and then tested on a ECG from the test data. The ECG that were explained by the model explainer were from a patient with non-specific intra ventricular conduction disorder (NSIVCB). The explaination is visualized in figure 3.

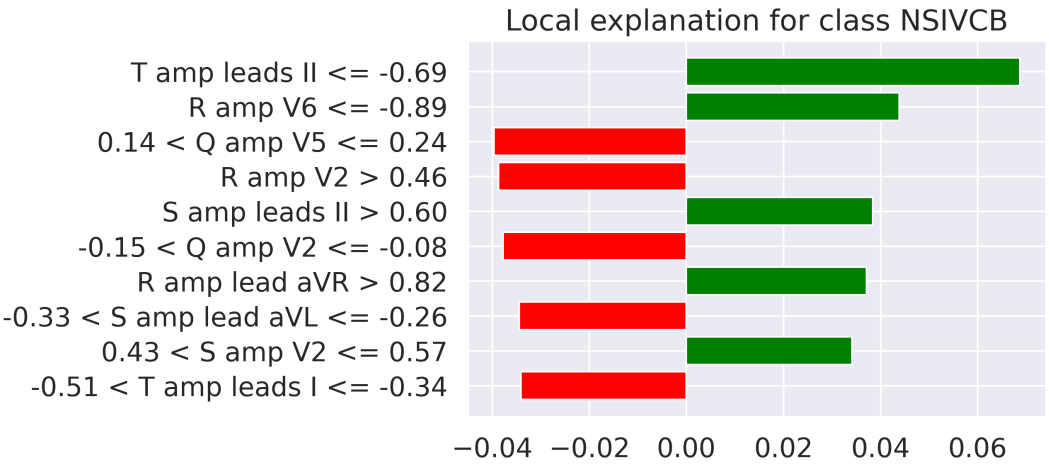


Figure 3: The figure shows the top 10 features from a ECG that contributed with the prediction. In the labels on the vertical axis; amp is short for amplitude, P, Q, R,S,T are the characteristic peaks in the ECG and I, II, III, aVL, aVF, aVR, V1, V2, V3, V4, V5 and V are the name of the 12 ECG leads. The ECG-features are extracted from a ECG from a patient with non-specific intra ventricular conduction disorder (NSIVCB). The green bars indicate the the degree of contribution towards an positive classification of NSIVCB, while the red bars shows the degree of contribution towards a negative classification of NSIVCB.

The recurrent explainer, used on the Encoder model, was trained on 5000 ECGs from the training data and then tested on a ECG from the test data. the ECG from the test data were abnormal and predicted correctly by the CNN model. The explaination model returned the channel/lead and the index of the sample/feature that were most important for the prediction by the Encoder. Figure 4 shows the tree most important features in lead aVR for the given test data.

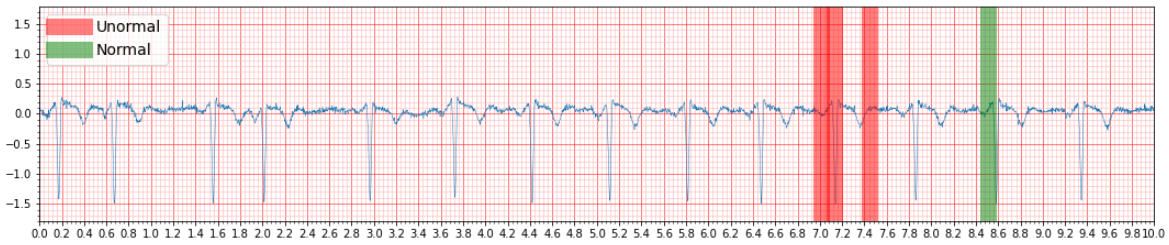


Figure 4: The figure shows a 10 seconds long ECG-recording from the aVR-lead. The ECG is correctly classified as abnormal by a 1D CNN Encoder. The horizontal, transparent green and red lines marks the features in the ECG that is seen as normal and abnormal by the recurrent explaination model. THe green line indicates the parts of the ECG that contributes towards a normal classification while the red line indicates the parts the ECG that contributes towards an abnormal classification.

4. Discussion

4.1. Cross-validated results

This paper demonstrates how CNN and ensemble models can be used to classify multiple cardiac abnormalities using 12-lead ECG-recordings. The cross-validated results show that the proposed ensemble model, using features from 12 leads, outperforms the other nine models on the development dataset from PhysioNet/CinC Challenge 2020. F1, F2, G2 and the PhysioNet/CinC Challenge Score was used to score the models. The 12-lead ensemble model were significantly better than all other models, measured on all four metrics.

The winner of PhysioNet/CinC Challenge 2020 reported a mean cross-validated PhysioNet/CinC Challenge Score of 0.533 ± 0.046 SD (28), which was slightly better than the best score achieved in this study: 0.512 ± 0.006. It is important to bear in mind that the cross-validated scores achieved on the development set, in this study, should be compared with caution to the reported scores on the PhysioNet/CinC Challenge 2020 hidden test set from other studies (28; 6). Even if some papers demonstrated good agreement between their cross-validated results, on the development set, and the results achieved on the hidden test set (28), the organizer reported that high-performing algorithms exhibited significant drops ($\approx 10\%$) in performance on the hidden test data (15).

Surprisingly, the CNNs with the lowest complexity performed best compared to the rest of the CNNs. The Encoder model were significantly better in terms of F1, F2 and G2-score (figure 2a-c), but closly follwed by FCN and FCN || Gender&Age when looking at the PhysioNet/CinC Challenge Score (Figure 2d). Nevertheless, it is stated in (6) that the Encoder performed worse than FCN || Gender&Age, Encoder || Gender&Age, Encoder || FCN + rule-based model and Encoder || FCN || Gender&Age + rule-based

model on a subset of the hidden test set. This observation emphasizes that one should be careful when comparing cross-validated scores with scores achieved on the hidden test set.

The FCN || Encoder and the FCN || Encoder || Gender&Age appeared to be unaffected by adding the rule-based model. No significant differences can be seen for any of the scoring methods in figure 2. A possible explanation for this might be that the rule-based model allways agreed with the CNN-model and thus did not change the prediction. Another possible explanation for this is that the rule-based model failed to analyze the ECG and then did not make any prediction. One should keep in mind that these rule-based algorithm were really simple and therefore the these results should be interpreted with caution.

Another surprising aspect was that the ensemble model, using features from 2 leads, performed significantly better than all CNN models, using all 12 leads, on the PhysioNet Challenge Score (figure 2d). However, it should be mentioned that the Encoder performed equally well on the F1, F2 and G2-score as seen in figure 2a-c.

A possible limitation in this study is that the ECGs were not filtered before feeding them into the model or before extracting features with the ECG-featurizer. Some of the ECGs showed a lot of noise, like the ECG in figure 5. Further studies are needed to determine if a filtered ECG signal would improve the performace of the models used in this study.

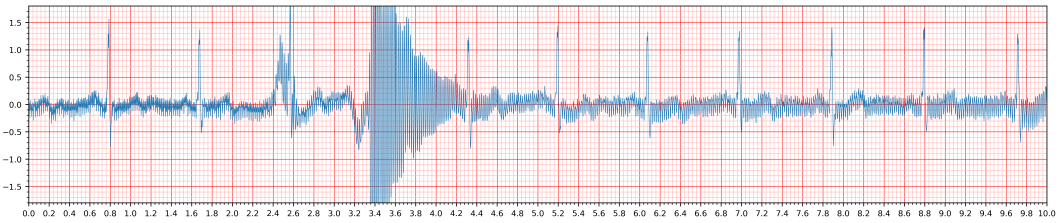


Figure 5: The figure shows a noisy ECG-signal from ECG Lead II

Limitation regarding LIME vs SHAP

Our findings emphasize the need to continue to develop explainability models for time series classifiers.

The 84 unscored diagnoses that were overlooked but still represented noise in the dataset.and did not represent any change in the 27-bit long label arra

It might be possible to uncover new features in the ECG that cardiologists should consider when interpreting ECG from athletes.

5. Conclusions

This study aimed to compare ten models and their feasibility to classify study aim to has reported that a ensemble model using features extracted from the ECG were able to outperform

References

[1] Cardiovascular diseases (CVDs). URL https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds).

[2] of Health and Human Services, U. D. National Ambulatory Medical Care Survey: 2015 State and National Summary Tables (2015). URL https://www.cdc.gov/nchs/data/ahcd/namcs_summary/2015_namcs_web_tables.pdf.

[3] Bickerton, M. & Pooler, A. Misplaced ECG Electrodes and the Need for Continuing Training. *British Journal of Cardiac Nursing* **14**, 123–132 (2019).

[4] Schläpfer, J. & Wellens, H. J. Computer-Interpreted Electrocardiograms. *Journal of the American College of Cardiology* **70**, 1183–1192 (2017).

[5] Smulyan, H. The Computerized ECG: Friend and Foe. *The American Journal of Medicine* **132**, 153–160 (2019).

[6] Singstad, B.-J. & Tronstad, C. Convolutional Neural Network and Rule-Based Algorithms for Classifying 12-lead ECGs 4.

[7] Annam, J. R., Kalyanapu, S., Ch., S., Somala, J. & Raju, S. B. Classification of ECG Heartbeat Arrhythmia: A Review. *Procedia Computer Science* **171**, 679–688 (2020). URL http://www.sciencedirect.com/science/article/pii/S1877050920310425.

[8] Mathews, S. M., Kambhamettu, C. & Barner, K. E. A novel application of deep learning for single-lead ECG classification. *Computers in Biology and Medicine* **99**, 53–62 (2018).

[9] Liu, S.-H., Cheng, D.-C. & Lin, C.-M. Arrhythmia Identification with Two-Lead Electrocardiograms Using Artificial Neural Networks and Support Vector Machines for a Portable ECG Monitor System. *Sensors (Basel, Switzerland)* **13**, 813–828 (2013). URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3574706/.

[10] Ribeiro, A. H. *et al.* Automatic diagnosis of the 12-lead ECG using a deep neural network. *Nature Communications* **11**, 1760 (2020). URL https://www.nature.com/articles/s41467-020-15432-4. Number: 1 Publisher: Nature Publishing Group.

[11] Yao, Q., Wang, R., Fan, X., Liu, J. & Li, Y. Multi-class Arrhythmia detection from 12-lead varied-length ECG using Attention-based Time-Incremental Convolutional Neural Network. *Information Fusion* **53**, 174–182 (2020). URL http://www.sciencedirect.com/science/article/pii/S1566253518307632.

[12] Li, D., Wu, H., Zhao, J., Tao, Y. & Fu, J. Automatic Classification System of Arrhythmias Using 12-Lead ECGs with a Deep Neural Network Based on an Attention Mechanism. *Symmetry* **12**, 1827 (2020). URL https://www.mdpi.com/2073-8994/12/11/1827. Number: 11 Publisher: Multidisciplinary Digital Publishing Institute.

[13] Chen, T.-M., Huang, C.-H., Shih, E. S., Hu, Y.-F. & Hwang, M.-J. Detection and Classification of Cardiac Arrhythmias by a Challenge-Best Deep Learning Neural

Network Model. *iScience* **23**, 100886 (2020). URL https://linkinghub.elsevier.com/retrieve/pii/S2589004220300705.

[14] Classification of 12-lead ECGs: the PhysioNet/Computing in Cardiology Challenge 2020. URL https://physionetchallenges.github.io/2020/.

[15] Alday, E. A. P. *et al.* Classification of 12-lead ECGs: the PhysioNet/Computing in Cardiology Challenge 2020. *Physiol. Meas.* (Under Review) (2020).

[16] Goldberger, A. L. & et al. PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals. *Circulation* **101** (2000).

[17] Liu, F. *et al.* An Open Access Database for Evaluating the Algorithms of Electrocardiogram Rhythm and Morphology Abnormality Detection. *Journal of Medical Imaging and Health Informatics* **8**, 1368–1373 (2018).

[18] Bousseljot, R., Kreiseler, D. & Schnabel, A. Nutzung der EKG-Signaldatenbank CARDIODAT der PTB über das Internet. *Biomedizinische Technik/Biomedical Engineering* 317–318 (2009). URL https://degruyter.com/view/j/bmte.1995.40.issue-s1/bmte.1995.40.s1.317/bmte.1995.40.s1.317.xml.

[19] Wagner, P., Strodthoff, N., Bousseljot, R., Samek, W. & Schaeffter, T. PTB-XL, a large publicly available electrocardiography dataset (version 1.0.1). PhysioNet. (2020).

[20] Pedregosa, F. & et al. Scikit-Learn: Machine Learning in Python. *Journal of Machine Learning Research* **12**, 2825–2830 (2011).

[21] Bjørn-Jostein Singstad. ECG-featurizer. URL https://doi.org/10.5281/ZENODO.4265151.

[22] Bjørn-Jostein Singstad & Tronstad, C. Classifying 12-Lead ECG Using Convolutional Recurrent Neural Network (2020).

[23] Szymanski, P. & Kajdanowicz, T. Scikit-multilearn: a scikit-based Python environment for performing multi-label classification. *The Journal of Machine Learning Research* **20**, 209–230 (2019).

[24] Szymański, P., Kajdanowicz, T. & Kersting, K. How Is a Data-Driven Approach Better than Random Choice in Label Space Division for Multi-Label Classification? *Entropy* **18** (2016). URL http://www.mdpi.com/1099-4300/18/8/282.

[25] Read, J., Pfahringer, B., Holmes, G. & Frank, E. Classifier chains for multi-label classification. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 254–269 (Springer, 2009).

[26] Nelder, J. A. & Mead, R. A Simplex Method for Function Minimization. *The Computer Journal* **7**, 308–313 (1965).

[27] Virtanen, P. & et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods* **17**, 261–272 (2020).

[28] Natarajan, A. *et al.* A Wide and Deep Transformer Neural Network for 12-Lead ECG Classification 4.

Convolutional Neural Network and Rule-Based Algorithms for Classifying 12-lead ECGs

Bjørn-Jostein Singstad [1], Christian Tronstad [2]

[1]University of Oslo, Oslo, Norway
[2] Oslo University Hospital, Oslo, Norway

Abstract

The objective of this study was to classify 27 cardiac abnormalities based on a data set of 43101 ECG recordings. A hybrid model combining a rule-based algorithm with different deep learning architectures was developed.

We compared two different Convolutional Neural Networks, a Fully Convolutional Neural Network and an Encoder Network, a combination of both, and with the addition of another neural network using age and gender as input. Two of these combinations were finally combined with a rule-based model using derived ECG features. The performance of the models was evaluated on validation data during model development using hold-out validation. Finally, the models were deployed to a Docker image, trained on the provided development data, and tested on the Challenge validation set. The model that performed best on the Challenge validation set was then deployed and tested on the full Challenge test set. The performance was evaluated based on a particular Challenge score.

Our team, TeamUIO, achieved a Challenge validation score of 0.377, and a full test score of 0.206 for our best model. The score on the full test set placed us at 20th out of 41 teams in the official ranking.

1. Introduction

The ECG reflects the electrical activity of the heart, and the interpretation of this recording can reveal numerous pathologies of the heart. An ECG is recorded using an electrocardiograph, where modern clinical devices usually contain automatic interpretation software that interprets the ECGs directly after recording. Although automatic ECG interpretation started in the 1950s, there are still some limitations [1, 2]. Because of the errors done by the automatic interpretation software, doctors have to read over the ECGs [3]. This is time-consuming for the doctors and requires a high degree of expertise [4]. There is clearly a need for better ECG interpretation algorithms.

Recent years have shown a rapid improvement in the field of machine learning. A sub-field of machine learning is called deep learning, where more complex architectures of neural networks are better able to scale with the amount of data in terms of performance. This type of machine learning has shown promising performance in many fields including medicine, and in this study, we have explored the usefulness of deep learning in classifying 12-lead ECGs.

As a starting point for our model architecture, we chose to use the two best performing Convolutional Neural Networks (CNN) used on ECG data in Fawaz HI et al 2019 [5]. They reported that Fully Convolutional Neural Networks (FCN) outperformed eight other CNN architectures compared. We also wanted to test the second-best architecture from their study which was an Encoder network. Finally, we assessed the integration of a rule-based algorithm within these models to test the performance of a CNN and rule-based hybrid classifier.

This study is a part of the PhysioNet/Computing in Cardiology (CinC) Challenge 2020, where the aim was to develop an automated interpretation algorithm for the identification of multiple clinical diagnoses from 12-lead ECG recordings.

2. Methods

2.1. Data

To train the CNN models a data set containing 43.101 ECG recordings with corresponding information files describing the recording, patient attributes, and the diagnosis was used [6, 7]. The recording lengths varied across the different ECG signals, 83.4% were 5000 samples long. 98.5% of the recordings were sampled at a frequency of 500Hz, 1.3% signals sampled at 1kHz and 0.2% signals sampled at 257Hz.

2.2. Preprocessing

According to the goal of this Challenge, we aimed to classify 27 of the 111 diagnoses [6]. The 27 labels to classify were One-Hot encoded, with each diagnosis rep-

resented as a bit in a 27-bit long array. All recordings were padded and truncated to a signal length of 5000 samples. Padding and truncation were done by removing any parts longer than 5000 samples and adding a tail of $5000 - n$ zeros to any recording of length $n < 5000$.

2.3. CNN architectures

As a starting point for classifying the ECG-signals, we employed FCN and Encoder types of CNN models as described in Fawaz HI et al 2019 [5]. Two models were tested without any modifications to the architecture other than changing the input and output layers to fit our input data and output classes. All output layers of each model used a Sigmoid activation function.

To make use of the provided age and gender data, a simpler neural network model with 2 inputs, one hidden layer of 50 units, and 2 outputs in the final layer was added. This new model was combined with our FCN and Encoder models by concatenation of the last layer of the CNNs.

Age and gender data were passed into the simple neural network as integers, but in some information files, the age of the patient was not given and was assigned a value of -1. The gender data was transformed into integers, where a male was set equal to 0, female equal to 1, and unknown was set to 2.

The two CNN models (FCN and Encoder) were combined as parallel models, concatenated on the second last layer. This model was tested with and without a parallel dense layer[1].

2.4. Rule-based model

The rule-based algorithm used the raw ECG signal, without any padding or truncating, as input. R-peak detection [8], and heart rate variability (HRV) analysis was programmed to add relevant derived features to the algorithm. An HRV-score was obtained by computing the root mean square of successive differences between normal heartbeats (RMSSD) using the detected R-peaks as timing indicators of each heartbeat.

The rule-based algorithm was able to classify eight different diagnoses: atrial fibrillation, bradycardia, low QRS-complex, normal sinus rhythm, pacing rhythm, sinus arrhythmia, sinus bradycardia, and sinus tachycardia.

The rule-based algorithm performed classification independent of the deep learning models. If there was disagreement between the rule-based algorithm and the CNN model, the rule-based algorithm overwrote the classification from the CNN model.

[1]All models and algorithms are available here: `https://www.kaggle.com/bjoernjostein/physionet-challenge-2020`

2.5. Model development

The models were trained and validated on the development data using hold-out validation with a split of 90% for training and 10% for validation. The first fold in a stratified K-fold was used with a random seed of 42 [9]. The splitting was arranged such that the distribution of diagnoses was the same in both the train and validation data.

During training, the Area Under the Curve (AUC) score on the validation set was used to determine if the learning rate should drop or stay. The learning rate was initially set to 0.001 for all models and decreased by a factor of 10, using the reduce on plateau method [10], for each epoch that the AUC score did not improve. Early stopping [10] was triggered when the AUC score on the validation data did not improve over two successive epochs.

2.6. Threshold optimization

The prediction thresholds were optimized during model development. This was done by running the classifier on the hold-out validation data and receiving a score between 0 and 1 for each of the classes. The Nelder-Mead downhill simplex method [11, 12] was applied to optimize the threshold individually for the 27 classes. The Nelder-Mead downhill simplex method is used to find the local minimum of a function using the function itself and an initial guess of the variable of the function. The 27-element long array was optimized using the negative of the PhysioNet/CinC Challenge score [6]. To increase the possibility of finding the global maximum of the PhysioNet/CinC Challenge score, all elements in the 27-element long array was given a value of 1 and multiplied it with a variable that was given values from 0 to 1, with a step size of 0.05. The value that gave the highest PhysioNet/CinC Challenge score was used as the initial guess for the Nelder-Mead downhill simplex method.

2.7. Model deployment

To obtain a valid score in the PhysioNet/CinC Challenge we submitted the models to the PhysioNet/CinC committee for testing on a Challenge validation and test set [6].

A Docker image was used to create a virtual Python environment for the model to be tested. During model deployment, the model was trained on the whole development set. The first three Challenge validation scores were obtained using AUC on the development data to schedule the reduction of the learning rate.

The two last Challenge validation scores were obtained using a learning rate scheduler. The learning rate schedule was programmed to be the same as in model development.

Model ID and name	Rule-based model	AUC	F1	F2	G2	Challenge score
A) FCN	No	0.875	0.381	0.446	0.230	0.348
B) Encoder	No	0.866	0.396	0.429	0.228	0.398
C) FCN + age, gender	No	0.877	0.368	0.438	0.222	0.385
D) Encoder + age, gender	No	0.828	0.334	0.389	0.190	0.333
E) Encoder + FCN	No	0.872	0.399	0.436	0.237	0.409
F) Encoder + FCN	Yes	0.872	0.361	0.413	0.203	0.348
G) Encoder + FCN + age, gender	No	0.866	0.400	0.434	0.233	0.395
H) Encoder + FCN + age, gender	Yes	0.866	0.356	0.405	0.198	0.338

Table 1. Scores were obtained by eight different models during model development. The models were evaluated by five different metrics, AUC, F1, F2, G2, and the PhysioNet/CinC Challenge score, during model development.

2.8. General parameters for both validation and testing procedures

For all models in both development and deployment, we used Adam optimizer, a batch size of 30, and binary cross-entropy as the loss function. A batch generator was used to feed the model with data during training, programmed to shuffle the order of data for each epoch.

Weights based on the number of occurrences of the different classes were calculated to deal with the skewed classes in the development data [9]. The calculated weights were passed to the model during training to give higher priority to rare diagnoses and lower priority to diagnoses that occurred more frequently.

3. Results

3.1. Scoring metrics

During model development, all models were validated on a subset of the development data using the metrics AUC (Eq 1), F_1-score (Eq 2), F_2-score (Eq 3), G_2-score (Eq 4), and the PhysioNet/CinC Challenge score, as seen in Table 1. On the Challenge validation set, we only obtained the PhysioNet/CinC Challenge score as seen in Table 2. After the evaluation of the performance on the full Challenge test set we were provided AUC (Eq 1), F_1-score (Eq 2), PhysioNet/CinC Challenge score, an Area Under the Precision-Recall Curve (AUPRC) score, and an accuracy score.

$$AUC_{(t_i - t_{i-1})} = (t_i - t_{i-1}) \times \frac{f(t_i) + f(t_{i-1})}{2} \quad \text{(Eq 1)}$$

$$F_1 = \frac{2 \times TP}{2 * TP + FP + FN} \quad \text{(Eq 2)}$$

$$F_2 = \frac{(1 + 2^2) \times TP}{(1 + 2^2) \times TP + FP + 2^2 \times FN} \quad \text{(Eq 3)}$$

$$G_2 = \frac{TP}{TP + FP + 2 \times FN} \quad \text{(Eq 4)}$$

3.2. Classification performance

Five out of the eight models tested during the development phase, as seen in Table 1, were successfully deployed and obtained a score on the Challenge validation set, presented in Table 2.

Model ID and name	Rule-based model	Challenge score
B) Encoder	No	0.229
C) FCN + age, gender	No	0.302
D) Encoder + age, gender	No	0.272
F) Encoder + FCN	Yes	0.377
H) Encoder + FCN + age, gender	Yes	0.364

Table 2. The scores are obtained on the Challenge validation set and only the PhysioNet/CinC Challenge score was given. The Challenge validation set is a subset of the Challenge test set and not the final score in the challenge. The scores achieved on the Challenge validation set was used to select one model for deployment on the full Challenge test set.

The best score on the Challenge validation set was achieved by model H, an Encoder in parallel with an FCN with the rule-based algorithms added, as seen in Table 2. Model H was finally deployed and scored on the full Challenge test set [6]. The model achieved an AUC-score of 0.728, an F_1-score of 0.233, and a PhysioNet/CinC Challenge score of 0.206. This score brought us, TeamUIO, to 20th place in the PhysioNet/CinC Challenge 2020.

4. Discussion and conclusion

We chose to pad and truncate the signals to 5000 samples which were necessary to be able to feed the signal to

the CNN. The disadvantage was that some important information from segments of the ECG recordings might have been omitted in training the models. On the other hand, the derived features used in the rule-based implementation were based on complete recordings. Thus, the models that combined both CNN and rule-based algorithms used the entire signal when classifying the ECG.

Deployment of the models was done using two different ways of controlling the learning rate. The scores of models B, C, and D, on the Challenge validation set (Table 2), were obtained by using AUC on the development data to schedule the reduction of the learning rate. This might have contributed to overfitting indicated by the difference of the Challenge score of models B, C, and D in Table 1 compared with the same models in Table 2. The Challenge score achieved on the Challenge validation data by model F and G (Table 2), were obtained using a learning rate scheduler [10]. The PhysioNet/CinC Challenge scores achieved on the Challenge validation data by model F and G (Table 2) are more consistent with the PhysioNet/CinC Challenge score obtained on the development data in Table 1 for the same models. In summary, our result indicates that the models, deployed on the Challenge validation set, which kept the same training schedule as in the development model, seem to avoid overfitting and perform better on unseen data.

During the model development, we observed that the Encoder (model B) performed better than the FCN (model A) on the PhysioNet/CinC Challenge score as seen in Table 1. A plain FCN (model A) was not scored on the Challenge validation set and thus it remains unclear which of a plain FCN or a plain Encoder perform best on unseen data like the Challenge validation data.

The Encoder (model B) decreased in performance when a parallel model for age and gender was added (model D) during model development (Table 1). However, the performance increased when the Encoder (model B) was added a parallel model for age and gender (model D) when scoring the models on the Challenge validation set (Table 2). Based on the PhysioNet/CinC Challenge score, during model development (Table 1), the FCN (model A) improved in performance when adding a parallel model for age and gender (model C). However, we did not deploy a plain FCN (model A) to the Challenge validation set and thus it remains unclear if the FCN + age and gender (model C) would outperform the FCN (model A) on the Challenge validation set.

During model development (Table 1), the Encoder + FCN (model E) and the Encoder + FCN + age, gender (model G), decreased in performance when adding the rule-based model (model F and H). However, the PhysioNet/CinC Challenge score, achieved by model F and G on the Challenge validation set (Table 2), was better than the PhysioNet/CinC Challenge score achieved by the same models during model development (Table 1). Our results indicate that the hybridization of CNN with a rule-based model could improve the diagnostic classification of ECG, but further analysis is needed to confirm whether, and to which extent such implementation improves the performance of the proposed CNN models.

References

[1] Schläpfer J, Wellens HJ. Computer-Interpreted Electrocardiograms. Journal of the American College of Cardiology August 2017;70(9):1183–1192.

[2] Smulyan H. The Computerized ECG: Friend and Foe. The American Journal of Medicine February 2019;132(2):153–160.

[3] Alpert JS. Can You Trust a Computer to Read Your Electrocardiogram? The American Journal of Medicine June 2012;125(6):525–526.

[4] Bickerton M, Pooler A. Misplaced ECG Electrodes and the Need for Continuing Training. British Journal of Cardiac Nursing March 2019;14(3):123–132.

[5] Fawaz HI, et al. Deep Learning for Time Series Classification: A Review. Data Mining and Knowledge Discovery July 2019;33(4):917–963.

[6] Alday EAP, Gu A, Shah A, Robichaux C, Wong AKI, Liu C, Liu F, Rad AB, Elola A, Seyedi S, Li Q, Sharma A, Clifford GD, Reyna MA. Classification of 12-lead ECGs: the PhysioNet/Computing in Cardiology Challenge 2020. Physiol Meas 2020;(Under Review).

[7] Goldberger AL, et al. PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals. Circulation June 2000;101(23).

[8] Pan J, Tompkins WJ. A Real-Time QRS Detection Algorithm. IEEE Transactions on Biomedical Engineering March 1985;BME-32(3):230–236.

[9] Pedregosa F, et al. Scikit-Learn: Machine Learning in Python. Journal of Machine Learning Research 2011; 12(85):2825–2830.

[10] Martín Abadi, et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. 2015.

[11] Nelder JA, Mead R. A Simplex Method for Function Minimization. The Computer Journal January 1965;7(4):308–313.

[12] Virtanen P, et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. Nature Methods March 2020;17(3):261–272.

Address for correspondence:

Bjørn-Jostein Singstad
Sem Sælands vei 24, 0371 Oslo, Norway
b.j.singstad@fys.uio.no