

Nicky van Oorschot - TMOI-ML-22

De repository is te vinden op: [nicky508/nickyvanoorschot_mads_exam_24 \(github.com\)](https://github.com/nicky508/nickyvanoorschot_mads_exam_24)

Opdracht maak/hypertune een model voor het classificeren van heart samples. Explicit in de opdracht wordt vernoemd dat men liever te veel normale heart samples als abnormaal classificeert en dus liever een hogere recall heeft. Om dit te bewerkstelligen en ook toch een goede precisie te bereiken, wilde ik gebruik maken van 2 modellen een variational autoencoder, met een hogere recall en een 2^{de} model een gehypertunde versie van het al gebouwde 2d convolutional model om de abnormaal geclassificeerde heart samples opnieuw te beoordelen met een hogere precisie. Echter omwille van tijd heb ik enkel de autoencoder gebouwd.

Recurrent variational autoencoder

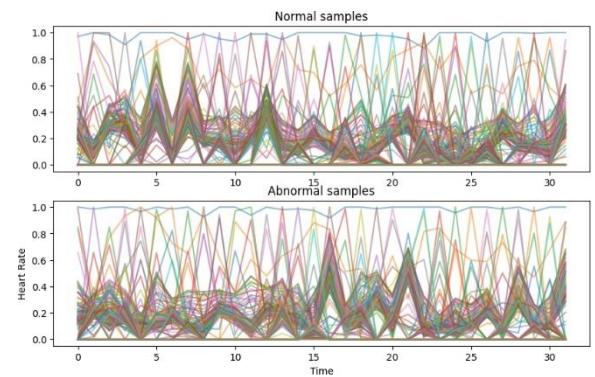
Voor de variational autoencoder (VAE) is gekozen om met twee LSTM-lagen te werken. De encoder vertaalt de timeserie met behulp van een LSTM-laag naar een hidden state, waarna de tweede LSTM-laag deze hidden state omzet naar de latent space. De decoder doet precies het tegenovergestelde: deze vertaalt de latent space terug naar de hidden state en reconstrueert de oorspronkelijke timeserie via een fully connected layer. De architectuur is te vinden in bijlage A. Het kiezen van de latent space, de hidden state, en de threshold waren de grootste uitdagingen van de VAE.

Normale en abnormale heartsamples

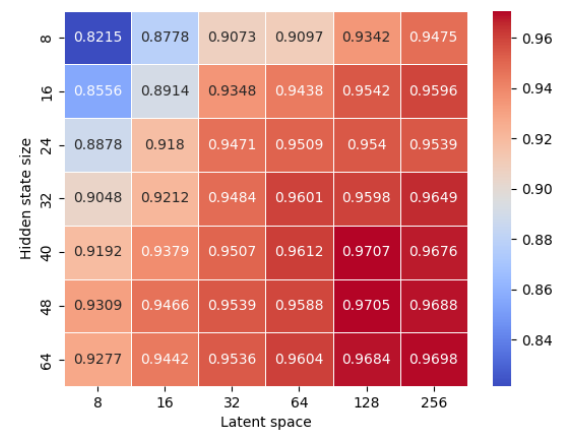
Als de normale en abnormale samples worden geplott, is duidelijk te zien dat er verschillende patronen in de data zitten. Dit betekent dat een autoencoder goed zou moeten kunnen werken (figuur 1). In beide sample sets is ook wat ruis te zien. De autoencoder wordt enkel getraind op de normale heartsamples, waarbij de autoencoder meer moeite zal hebben om de abnormale heartsamples te reconstrueren. De VAE is gehypertuned om de beste latent space, hidden state size en threshold te bepalen.

Latent space en hidden state size

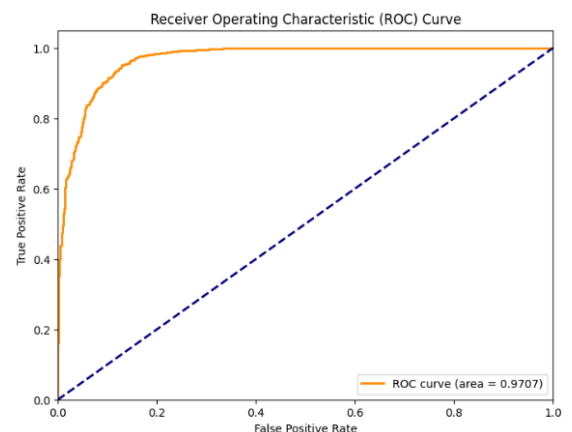
Om de juiste grootte van de latent space en de hidden state te bepalen, is een grid search uitgevoerd. Uit handmatige aanpassingen van verschillende groottes bleek dat de ideale latent space ergens tussen de 8 en 48 zou liggen, waarbij de hidden state size een grote invloed heeft. Tijdens de grid search zijn alle combinaties 200 epochs zijn getraind (met een early stopping van 10 epochs). De veelgebruikte metric accuracy is niet geschikt voor dit probleem, omdat de VAE geen directe classificatietaak uitvoert. Daarom is de ROC AUC (Area under the curve) metric gebruikt. Deze metric geeft, onafhankelijk van een specifieke drempelwaarde, de bekwaamheid van het model weer om abnormale van normale heart samples te onderscheiden. Dit resulteerde in de heatmap in figuur 2. De sweetspot lijkt rond de 40 voor de latent space te liggen met een hidden state van 128. Dit levert een ROC-AUC op zoals te zien in figuur 3. Het is duidelijk te zien dat de autoencoder complexere structuren kan vastleggen met een grotere hidden state. Wellicht dat de autoencoder nog beter werkt met nog grotere hidden state sizes, maar dit zou betekenen dat het trainen van het model langer zal duren (terwijl het nu al vrij lang duurt). De latent space wil ik bij voorkeur laag houden. Op basis van de gekozen latent space en hidden state size, heb ik de verdeling van de reconstruction losses van zowel de normale als de abnormale heart samples in beeld gebracht. Het is zichtbaar dat de normale



Figuur 1



Figuur 2



Figuur 3

heart samples beter gereconstrueerd kunnen worden met een lagere reconstruction loss in contrast met de abnormale samples. Ook is zichtbaar dat er een kleine overlap is tussen de reconstruction losses van de normale en abnormale heart samples (figuur 4).

Threshold

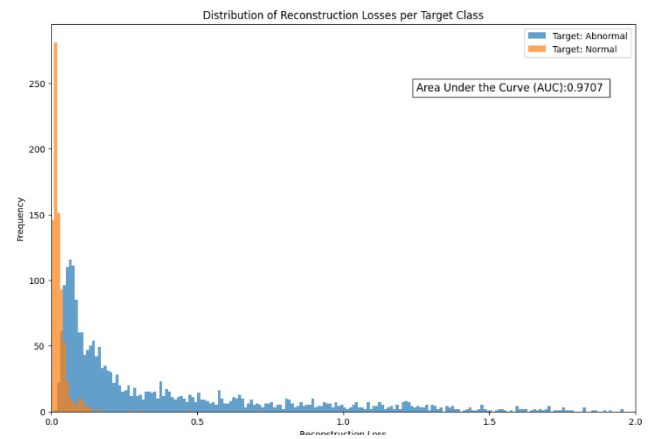
Nadat de ideale latent space en hidden state size zijn bepaald, is gezocht naar de best passende threshold. Voor de threshold in de range van 0.0 tot 0.2 zijn de recall, precision en F1-score berekend (bijlage B), en is gekeken naar de standaarddeviatie van de reconstruction loss van de normale heart samples. De standaarddeviatie is 0.0394. Dit betekent dat er ongeveer een precisie van 0.862 en een recall van 0.9857 is, wat leidt tot de confusion matrix in figuur 5.

Uiteindelijk worden er 81 anomalieën niet gevonden. In de opdracht is aangegeven dat men liever een hogere recall heeft, daarom heb ik in dit geval gekozen voor een iets lagere drempelwaarde, namelijk 0.03. Dit leidt tot een precisie van 0.821, een recall van 0.997 en een F1-score van 0.90. Dit resulteert in de confusion matrix in figuur 6. Hierdoor worden bijna alle abnormale heart samples gevonden, maar worden ook iets meer normale samples als abnormaal geïdentificeerd.

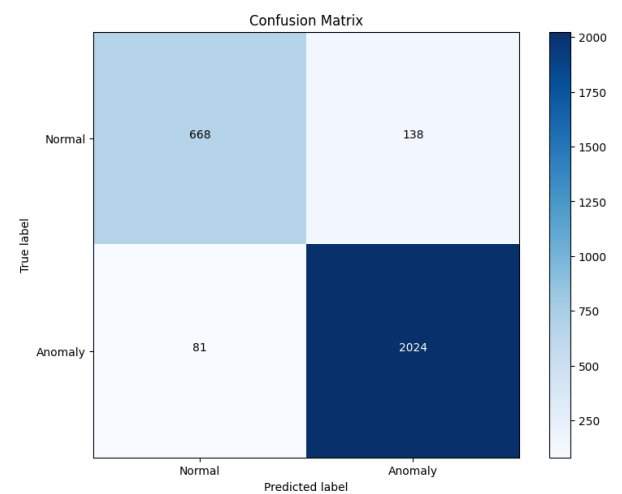
Vergelijkend met het aangeleverde Conv2d-model werden ruim 120 abnormale heart samples als normaal geïdentificeerd en daarmee gemist. 98 normale heart samples werden verkeerd geïdentificeerd, terwijl de autoencoder er ongeveer 261 verkeerd classificeert. De afweging in threshold zou in praktijk verder met de opdrachtgever gefinetuned kunnen worden.

Regulatie

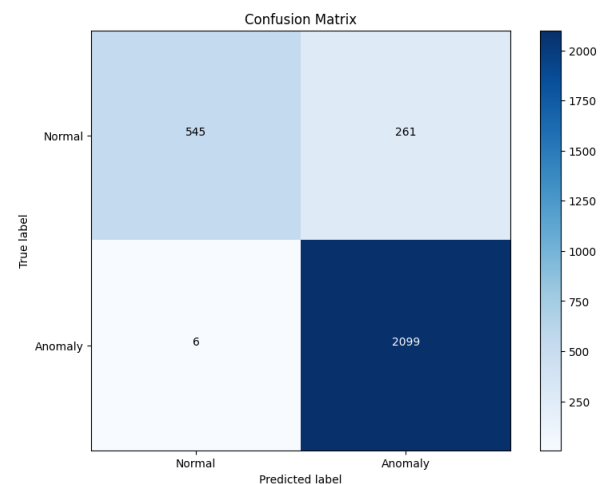
Om ervoor te zorgen dat de VAE beter in staat is om abnormale heart samples te onderscheiden, zijn verschillende regularisaties toegepast, zoals batchnormalisering, drop-outs en een aangepaste learning rate. Ik heb een drop-out van 0.2 toegepast om te voorkomen dat units overgespecialiseerd worden, zonder deze verder te hypertunen. Ik heb batchnormalisering geprobeerd, maar dit gaf ongewenste effecten. Door batchnormalisering was de autoencoder niet meer in staat om normale heart samples met een lagere reconstruction loss te reconstrueren dan abnormale samples. Ik denk dat dit mogelijk komt doordat de autoencoder belangrijke kenmerken leert en deze in de latent space stopt, en batchnormalisering vervormt mogelijk deze representatie in deze latent space. Daarnaast bleek bij het handmatig instellen van de autoencoder dat een learning rate van $1e-4$ beter werkte; het model kon langer leren zonder vroegtijdig te stoppen door de early stopping. Deze instelling heb ik behouden. In Bijlage C heb ik enkele reconstructies opgenomen.



Figuur 4

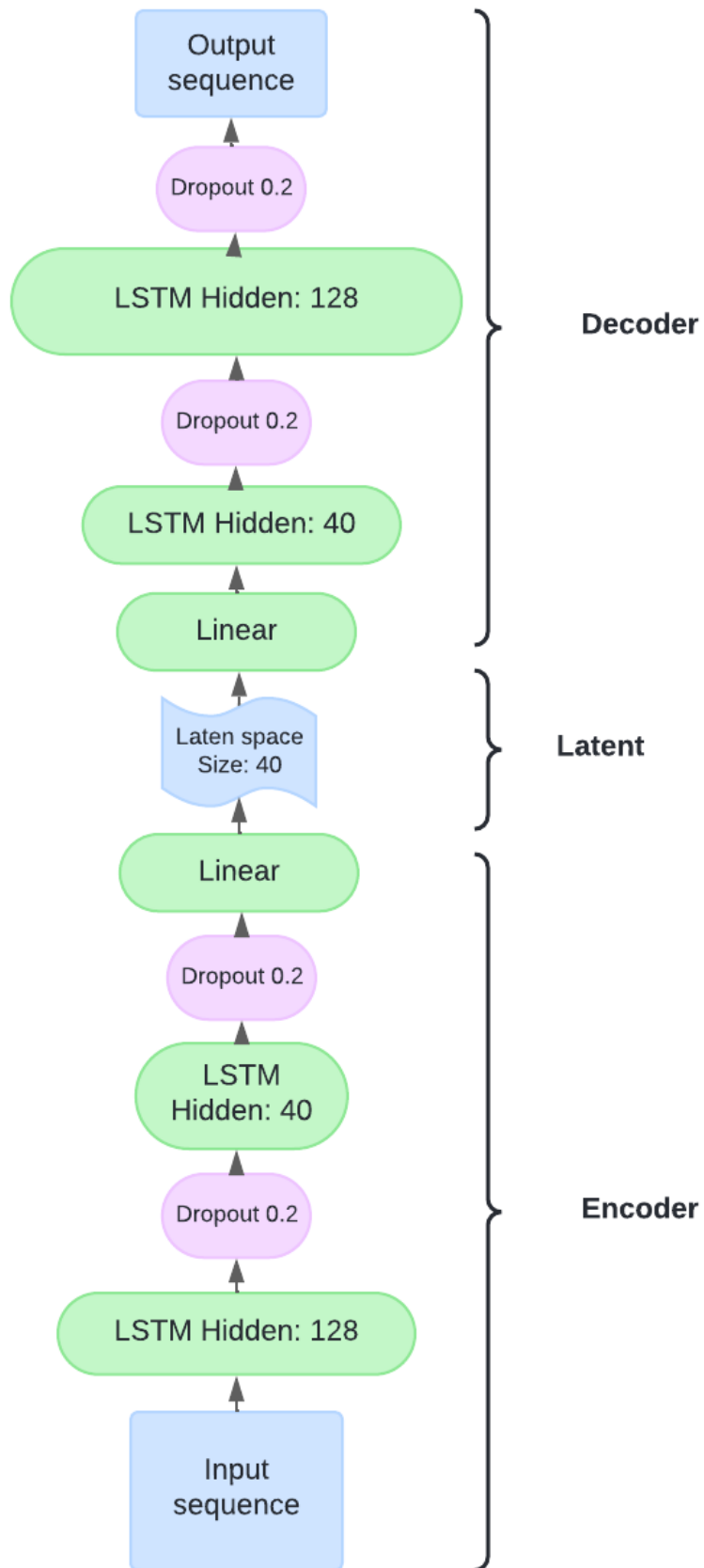


Figuur 5



Figuur 6

Bijlage A: Architectuur Autoencoder



Bijlage B: Thresholds

Threshold	Precision	Recall	F1 Score
0.00	0.7231	1.0	0.8393
0.01	0.7239	1.0	0.8398
0.02	0.7608	1.0	0.8641
0.03	0.8210	0.9976	0.9007
0.04	0.8628	0.9857	0.9202
0.05	0.8965	0.9715	0.9325
0.06	0.9277	0.9387	0.9332
0.07	0.9472	0.8950	0.9204
0.08	0.9571	0.8475	0.8990
0.09	0.9684	0.8143	0.8846
0.10	0.9737	0.7739	0.8624
0.11	0.9754	0.7335	0.8373
0.12	0.9780	0.6979	0.8145
0.13	0.9816	0.6599	0.7892
0.14	0.9845	0.6352	0.7722
0.15	0.9855	0.6147	0.7572
0.16	0.9857	0.5895	0.7378
0.17	0.9893	0.5715	0.7245
0.18	0.9915	0.5511	0.7084
0.19	0.9913	0.5411	0.7001
0.20	0.9911	0.5316	0.6920

Bijlage B: Reconstructies

