Source-agnostic gravitational-wave detection with recurrent autoencoders

IOP Publishing Mach. Learn.: Sci. Technol. 3 (2022) 025001 https://doi.org/10.1088/2632-2153/ac5435

MACHINE LEARNING
Science and Technology

OPEN ACCESS

RECEIVED
13 August 2021

REVISED
8 February 2022

ACCEPTED FOR PUBLICATION
9 February 2022

PUBLISHED
4 April 2022

PAPER

Source-agnostic gravitational-wave detection with recurrent autoencoders

Eric A Moreno[1,5], **Bartlomiej Borzyszkowski**[2,6], **Maurizio Pierini**[3,*], **Jean-Roch Vlimant**[4] and **Maria Spiropulu**[4]

[1] Massachusetts Institute of Technology, Cambridge, MA 02139, United States of America
[2] Gdansk University of Technology, Narutowicza 11/12, 80-233 Gdansk, Poland
[3] European Organization for Nuclear Research, 1211 Geneva 23, Switzerland
[4] California Institute of Technology, Pasadena, CA 91125, United States of America
[5] Previously at California Institute of Technology.
[6] Also at CERN and Intel Technology Gdansk, Poland.
[*] Author to whom any correspondence should be addressed.

E-mail: maurizio.pierini@cern.ch

Keywords: machine learning, unsupervised learning, anomaly detection

Abstract

We present an application of anomaly detection techniques based on deep recurrent autoencoders (AEs) to the problem of detecting gravitational wave (GW) signals in laser interferometers. Trained on noise data, this class of algorithms could detect signals using an unsupervised strategy, i.e. without targeting a specific kind of source. We develop a custom architecture to analyze the data from two interferometers. We compare the obtained performance to that obtained with other AE architectures and with a convolutional classifier. The unsupervised nature of the proposed strategy comes with a cost in terms of accuracy, when compared to more traditional supervised techniques. On the other hand, there is a qualitative gain in generalizing the experimental sensitivity beyond the ensemble of pre-computed signal templates. The recurrent AE outperforms other AEs based on different architectures. The class of recurrent AEs presented in this paper could complement the search strategy employed for GW detection and extend the discovery reach of the ongoing detection campaigns.

1. Introduction

The detection of gravitational waves (GWs) from stellar binaries such as black hole and neutron star mergers have ushered in a new era of analyzing the Universe. Operating together, the laser interferometer gravitational-wave observatory (LIGO) [1] and the Virgo Interferometer [2] can peer into deep space giving astronomers the ability to uncover and localize stellar processes through their gravitational signature. The first observation of a binary black hole merger (GW150914) [3] has given way to a plethora of GW events, and notably to the observation of intermediate-size black holes [4] and neutron star mergers [5], an event that marked the beginning of the multi-messenger astronomy era [6].

Instrumental on the software side of these observations are the algorithms which identify the faint GW signals in an environment characterized by overwhelming classical and quantum noise. The most sensitive detection algorithm, matched filtering (MF) [7], consists of matching incoming data with templates of simulated GW shapes, covering the parameter space of binary masses [8–12], which is then used to identify the signal. At the same time, they offer an estimate of the astrophysical parameters associated to the detected GW event, such as the nature and mass of the two merging objects. This method was extremely functional to the success of the LIGO and VIRGO observation campaigns. On the other hand, by relying on pre-computed templates, it could lead to missing events for which a template is not available. This could be an event originating from computationally prohibitive conditions [13–16], or even some sort of unforeseen GW source. To compensate for the limitations implicit in these assumptions, the data processing pipelines of

LIGO and VIRGO also included the coherent WaveBurst [17], an alternative strategy designed to deal with unmodeled GW sources.

Detecting GWs is certainly one of the hardest challenges faced in fundamental science in the recent years. Given how weak a GW signal is when compared to typical noise levels, it is natural to look at machine learning (ML), and especially to deep learning (DL), in order to improve a signal detection capability. For instance, an early attempt at multivariate classification with random forests [18] are furthered by [19–22] which discuss how Convolutional networks [23, 24] can be trained to extract a variety of GW signals from highly noisy data. This is an example of a supervised classifier, i.e. a classifier trained to separate different populations in data (e.g. signal vs. noise) by matching a given set of ground-truth labels . While classifiers could certainly contribute to enhance the current state-of-the-art detection capability, they rely on pre-defined signal much like the MF technique. In other words, they are designed to possibly improve the detection accuracy but they are not necessarily going to extend the detector sensitivity to exotic signals outside the portfolio of pre-simulated templates. In fact, these networks are typically trained on labeled data from simulation, so even in this case the capability of simulating a given signal is an underlying requirement. On the other hand, GW detection comes with the need of going beyond signal signatures that can be emulated. McGinn *et al* [25] discusses how to generate 'unmodeled' waveforms, which could then be used to train a supervised algorithm without the use of templates.

Besides the search for exotic sources, model independent strategies could be useful to deal with practical issues such as glitch detection [26].

In this paper, we investigate the possibility of rephrasing the problem of GW detection as an anomaly detection task. By anomaly detection we mean the use of a one-class classifier, in this case a Deep autoencoder (AE) [27], to identify outlier populations in an unlabeled dataset. An AE is a compression-decompression algorithm that is trained to map a given set of inputs into itself, by first compressing the input into a point in a learned latent space (encoding) and then reconstructing it from the encoded information (decoding).

Once trained on standard events, the AE might fail to reconstruct samples of different kind (the anomalies). Any input-to-output distance measurement can then be used to identify these anomalies. Under the assumption that these anomalies are rare, one can directly train these AEs on data, looking for the set of AE parameter values that minimize the difference between the input and the output, using some distance $\mathcal{D}$ as a loss function. By taking as a reference the distribution of $\mathcal{D}$ on data, one can label anomalies as the outlier events of this distribution. This very same approach was recently discussed in [28], where the discovery reach of Convolutional AEs for GW detection is investigated. In this work, we consider two recurrent AE architectures: long-short memory networks (LSTMs) [29], and gated recurrent units (GRUs) [30]). For comparison, we consider alternative AE architectures: dense (i.e. fully connected) neural networks (DNNs) and convolutional neural networks (CNNs).

The main advantage of this unsupervised strategy is that one algorithm is potentially sensitive to multiple signal typologies. On the other hand, this gain in flexibility is typically followed by a loss in accuracy. For a specific signal, an algorithm trained with an unsupervised procedure on unlabeled data is typically less accurate than a supervised classifier trained on labeled data.

The proposed strategy comes with another remarkable advantage: under the assumption that anomalies are rare instances in the processed data, AEs can be trained directly on data, without relying on signal or noise simulation. Instead, supervised algorithms require labels which, in the case of rare signals like those considered in this study, are obtained using synthetic data (e.g. from Monte Carlo simulation). Assuming this training could happen in real time, the AE could adapt to changing experimental conditions and limit the occurrence of false claims.

This paper is organized as follows: related works describing ML approaches to GW detection are briefly discussed in section 2. The dataset utilized for this study is described in section 3. The AE architectures are described in section 4, together with the corresponding classifiers used to benchmark performance. Results are presented in section 5. Conclusions are given in section 6.

2. Related work

Most DL approaches for GW classification [18–22, 31–33] involve supervised learning techniques, which typically provide competitive accuracy by exploiting network non-linearity and the information provided by ground-truth labels. By construction, these methods rely on a realistic simulation of the signal induced by a specific kind of source, which is assumed upfront.

In addition, Bayesian inference libraries [34–36] have been created to estimate the properties of a gravitational-wave source. Unsupervised anomaly detection of transients has also been proposed using a temporal version of k-nearest neighbors [37].

Principal component analysis [38], which performs a linear orthogonal transformation of a set of (possibly correlated) variables into a set of linearly uncorrelated variables, has also been introduced for transient detection [39, 40]. This can give a quick characterization of the intrinsic properties of a data sample. The DL methods discussed in this paper are generalizations of this approach to include nonlinear compression.

Boosting Neural Networks [41], which use a combination of unsupervised and supervised learning techniques, have also been used for GW classification [42]. This method performs an unsupervised hierarchical clustering on the incoming data to identify possible groups and a supervised Bayesian classifier to do the final classification. BNNs can classify a number of different injections including Gaussian, Ringdowns, Supernovae, white noise bursts, mergers, etc. Importantly, the architecture requires sufficient statistics to cluster in an unsupervised manner, which makes it not ideal for the processes that we focus on in this paper.

Finally, CNNs have achieved high accuracy while having the added benefit of parameter estimation to infer useful parameters of the GWs such as masses and spins of the binary merger components [19, 20]. In a similar approach to the one discussed in this paper, Morawski *et al* [28] discusses the use of CNN AEs in order to classify GWs. For comparison, the same architecture as specified in [28] is implemented in section 4 alongside the recurrent AEs.

3. Data samples

This study is performed on a sample of synthetic data, generated for this study and available on Zenodo [44, 45]. Data are generated using publicly available LIGO software suites [46, 47] and processed using the GGWD library [43].

Noise events occur when no signal is overlapped to the detector noise. Detector noise is generated with an aLIGOZeroDetHighPower [46, 48] power spectral density (PSD), which is computed as a function of the length of noise, time step of the noise, and noise weighting to color the noise. This is done using PyCBC [47]. This approach to simulated data generation ignores glitches, blips, and other transient sources of detector noise[7].

In absence of exotic signal sources, we use traditional GW signals to assess the detection performance. We consider two kinds of GW sources from binary mergers: binary black hole (BBH) and binary neutron star (BNS). Signal events are generated simulating GW production from compact binary coalescences using PyCBC [47], which itself uses algorithms from LIGO's LAL Suite [46]. Signal event containing GWs were created overlaying simulated GWs on top of detector noise. This provides an analogous situation to a real GW, in which the strain from the incoming wave is recorded in combination with the normal detector noise.

The dataset consists of 400 000 noise samples. Each sample corresponds to 8 s of data, sampled at 2048 Hz. We consider a LIGO-like experimental setup, with two detectors (L1 and H1) taking data simultaneously at different locations. The simulation also includes a difference in GW time-of-arrival at each detector due to light travel time between them, which is significant at a sampling rate of 2048 Hz. Within a signal 8 s event, the PSD estimate is used to calculate the network optimal MF signal-to-noise ratio (SNR), which is then used to scale the added injection to form a uniform SNR sample distribution. For each detector, a sample is represented as a one-dimensional array with 16 384 entries.

Figure 1 shows two of the simulated signal events: a coalescence of two black holes, with masses set to 66 and 34 solar masses ($M_\odot$), and a coalescence of two neutron starts, with masses set to 1.1 and 1.4 solar masses ($M_\odot$), detected by the L1 and H1 detectors. The generation parameters are listed in the figure: the spin of the two black holes, right ascension, declination, coalescence phase, inclination, polarization, and SNR [43]. In absence of any source of noise, the signal would appear as shown by the orange line. Once the noise is added, the detectable signal corresponds to the blue line.

This dataset is split in three parts: 256 000 training samples (64%), 64 000 validation samples (16%), and 80 000 test samples (20%). The training and validation datasets are used in the optimal-parameter learning process, while the test dataset is used to assess the algorithm performance after training, together with the signal samples.

The BBH sample is generated with the following parameters and priors:

- SEOBNRv4 [49] Approximant.
- Masses independently and uniformly varied within [10, 80] $M_\odot$.

[7] For this reason, the considered AEs could be re-purposed for anomaly detection algorithms to identify detector glitches. The main difference between these kind of glitch anomalies and those of astronomical significance would be the lack of coincidence of anomalies across different detectors, a clear indication of an anomalous signal of astrophysical origin.

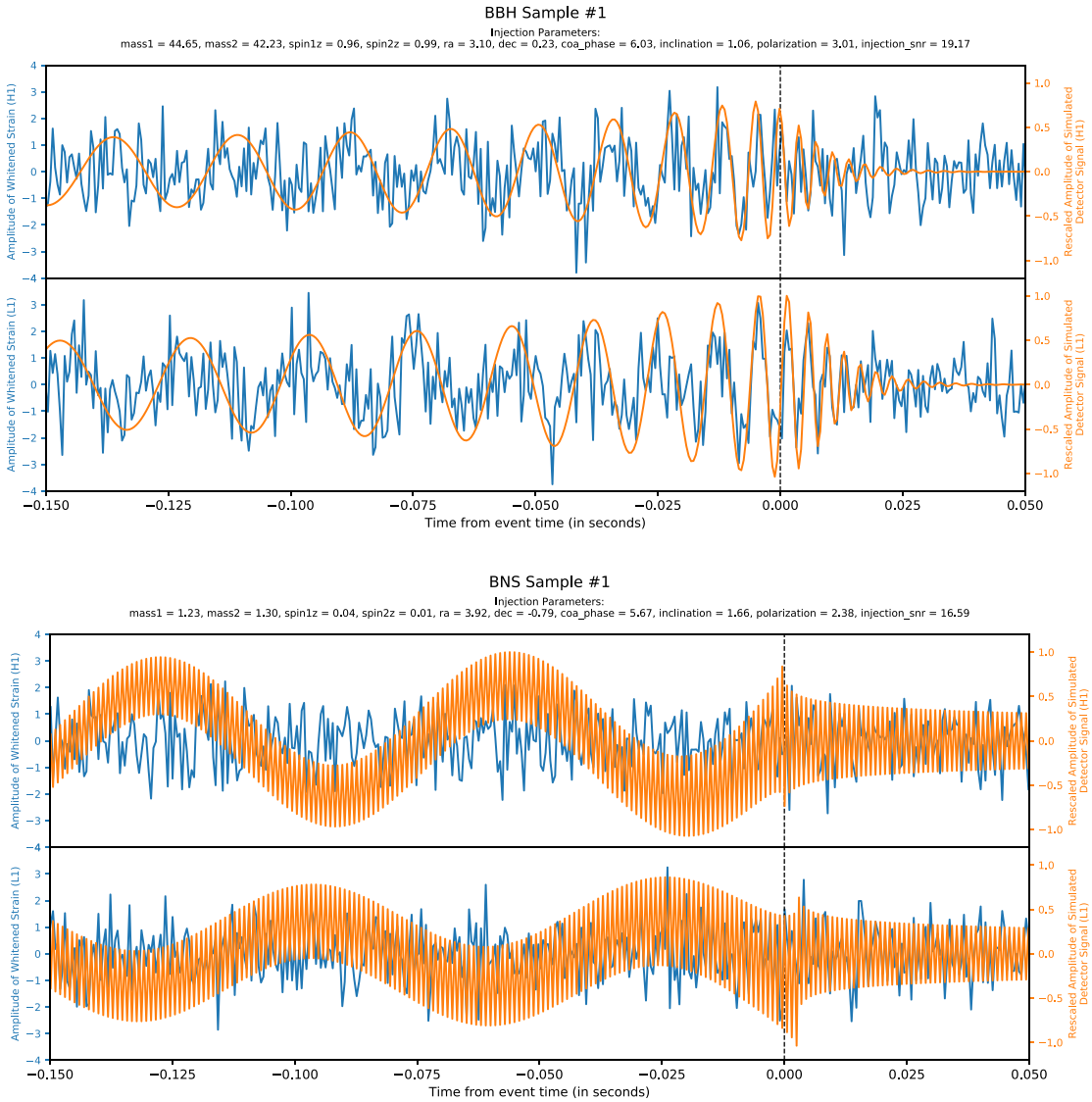

Figure 1. Resulting strains from simulated GWs on the Livingston (L1) and Hanford (H1) detectors from the coalescence of a 45 $M_\odot$ and 42 $M_\odot$ black holes (top) and 1.2 $M_\odot$ and 1.3 $M_\odot$ neutron stars (bottom), obtained using the GGWD package [43]. The amplitude of the GW signal, which has been arbitrarily scaled to be more visible, is illustrated in orange and the strain on the detector including this signal is illustrated in blue. As the amplitudes are arbitrarily scaled in this figure, it is important to note that binaries with lower mass will have longer waveforms and, for the same integrated SNR, will have a smaller instantaneous amplitude.

- Spins independently and uniformly varied within [0, 0.998].
- Injection network SNR uniformly varied within [5, 20] for full 8 s event.
- Coalescence phase uniformly varied within $[0, 2\pi]$.
- Inclination varied with Sine prior from $[0, \pi]$.
- Right Ascension sampled uniformly from $[0, 2\pi]$ using uniform_sky prior [47].
- Declination sampled uniformly from $[-\pi/2, \pi/2]$ using uniform_sky prior [47].
- Polarization sampled uniformly from $[0, 2\pi]$.

The BNS sample is generated with the following parameters and priors:

- IMRPhenomDNRTidal_v2 [50] Approximant.
- Masses independently and uniformly varied within [1.1, 2.1] $M_\odot$ [51].
- Spins independently and uniformly varied within [0, 0.05] [52].
- Injection network SNR uniformly varied within [5, 20] for full 8 s event.
- Coalescence phase uniformly varied within $[0, 2\pi]$.
- Inclination varied with Sine prior from $[0, \pi]$.
- Right Ascension sampled uniformly from $[0, 2\pi]$ using uniform_sky prior [47].

- Declination sampled uniformly from $[-\pi/2, \pi/2]$ using uniform_sky prior [47].
- Polarization sampled uniformly from $[0, 2\pi]$.

Data are whitened with a Fast Fourier Transform integration length of 4 s and a duration of the time-domain Finite Impulse Filter whitening filter of 4 s to remove the underlying correlation in the data [53]. Then, band-pass filtering was applied to remove high frequency (above 2048 Hz) and low frequency (below 30 Hz) components from the data. Doing so, background from outside the current interferometer sensitivity range is discarded. To facilitate the data processing and learning by the network, the data are scaled absolutely to a [0, 1] range. Each 8 s event is then cropped to 2.5 s around the GW event, with the GW happening at a random time after the beginning of the time window. The GW arrival is uniformly sampled in a [1, 2] s interval. This choice allows us to take into account the appropriate time for the signal ring-up/ring-down. This is done to assure that the model classifiers are not biased to a certain time period within the event, which would occur if the simulated data has GWs appearing at only a single time within the event window.

4. Network architectures

AEs are algorithms that project an input sample $x \in \mathcal{X}$ to its encoded projection z in a latent space $\mathcal{Z}$, typically of lower dimension than the input space. The encoded projection z is then decoded to a reconstructed $\hat{x} \in \mathcal{X}$. The network parameters determine how x is projected to z and then back to $\hat{x}$. Their values are fixed minimizing some input-to-output distance, used as a loss function in the network training. In this study, we consider the mean-squared error (MSE) between each element of the input array and the corresponding output. We consider several network architectures, all structured according to a common scheme with the decoder mirroring as close as possible the encoder architecture. Three specific architectures are introduced, with CNN, LSTM, or GRU layers.

As a comparison to recurrent layers, we implement a 1D CNN AE similar to [28] with an input of 25 one-dimensional windows of shape (1024, 1) slid around a 2.5 s interval sampled at 2048 Hz. Each of these 25 inputs are individually inputted into the AE, producing 25 reconstructed steps (and losses) to work with in post-processing analysis. The encoder consists of two one-dimensional convolutional layers with filters of size [256, 128] and kernel of size 3, coupled with a maxpool layer of size 2. The decoder mirrors this architecture with an upsampling layer of size 2, and two one-dimensional convolutional layers with filter size [256, 1] and kernel of size 3. In addition, DNN AEs with number of nodes 100, 50, 10 and 10, 50, 100 were attempted but yielded worse results. This is expected as DNN AEs are not specialized in time-series data, unlike recurrent architectures.

The LSTM and GRU networks function similarly, utilizing LSTM and GRU cells instead of simple CNN/DNN nodes. The latent space bottleneck in this representation is created by instructing the LSTM (or GRU) cells to only return their final state in the encoding phase and then repeating that final state as a vector which can then be input to the decoder. The input to these recurrent architectures are 256 one-dimensional arrays of shape (100, 1) slid around a 2.5 s interval sampled at 2048 Hz. Each of these 256 inputs are individually inputted into the AE, producing 256 reconstructed steps (and losses) to work with in post-processing analysis. The encoder consists of two layers, with number of units 32 and 8, which produces a latent representation by only outputting the final LSTM state vector on the bottleneck layer. The decoder consists of two layers with 8, 32 nodes, which is then multiplied by a final temporal slice of a dense layer, yielding the same dimensions as the input representation. For illustration, the architecture of the LSTM model is shown in figure 2.

The CNN, LSTM, and GRU AEs are all trained on unlabeled detector noise data, with no introduction to signal distributions. Doing so, the latent space representation returned by the encoder is exclusively a function of detector noise. As a result, the MSE error is relatively consistent during periods with exclusively noise, but it might instantaneously increase when a signal event passes through the AE. An example of such a spike in the MSE loss as a function of time is shown in figure 3. The spike is typically due to the fact that the encoding/decoding sequence learned on noise might not be optimal for a previously unobserved kind of input data. As a result, the distance between the input and the output could be larger for anomalous data, up to generate a spike. Operationally, one could then monitor the MSE value returned by the algorithm, and the detection of a signal could be correlated to the observation of a spike above threshold.

The performance of the three AE models is assessed comparing their accuracy on benchmark signal samples to that obtained from binary CNN classifiers, trained on the same data and the corresponding labels. Different classifiers are trained for different signal topologies. The classifier architecture is loosely equivalent to that of the CNN encoder from [19, 20]. In particular, it consists of four convolution layers, with 16, 32, 64, and 128 filters respectively, and three fully connected layers with 128, 64, and 2 nodes, respectively. A ReLU

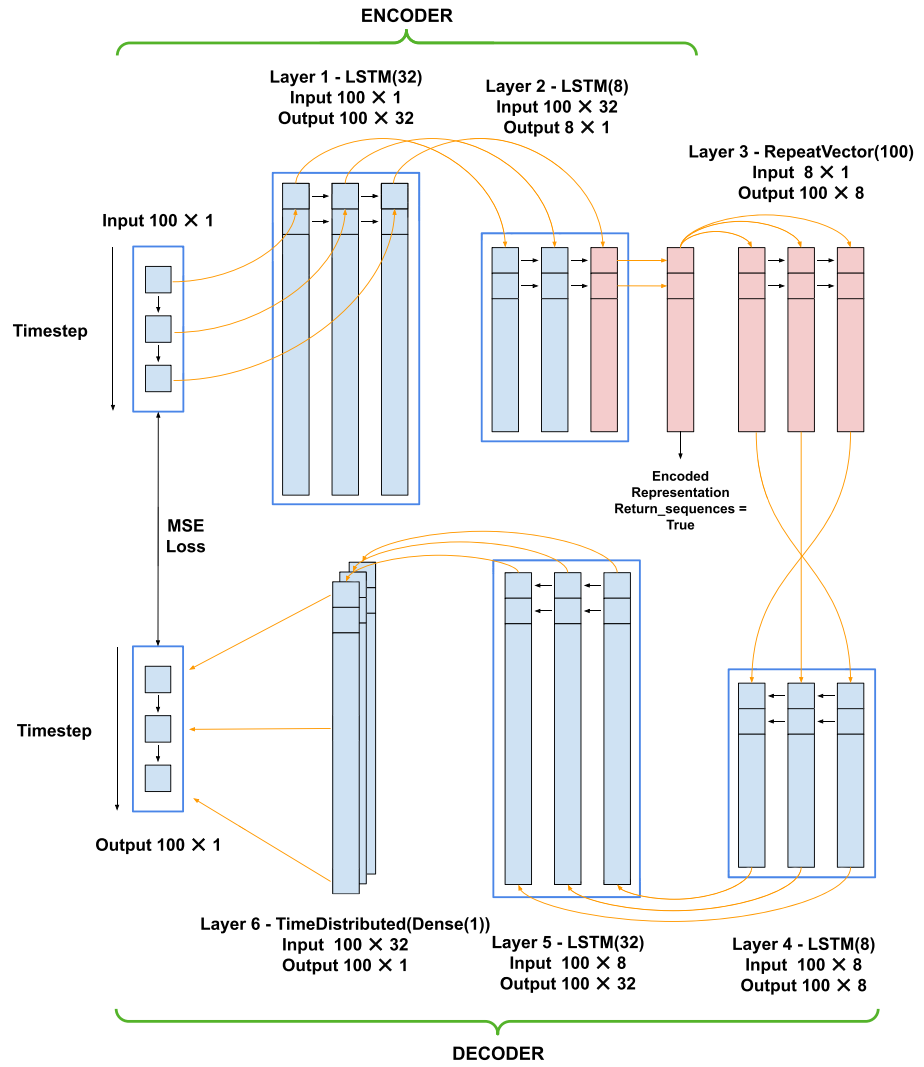

Figure 2. Graphical illustration of LSTM AE. The general encoder structure still remains, and the latent space is created by setting return_sequences = false in the LSTM cell, passing through only the final time step of the LSTM. This latent space is then repeated using a RepeatVector, forcing the encoder to create some important latent representation as its final output which can then be sent to the decoder. A similar workflow can be established for a GRU AE or any other recurrent AE.

activation function was used throughout. Kernel sizes of 4, 4, 4, and 8 were used with dilation rates of 1, 2, 2, and 2 for the convolutional layers and kernel sizes of 2, 2, 4, 4, with a stride of 4 for all the max pooling layers. A sigmoid function is used for the single-node output layer. An LSTM-implementation of the supervised classifier was also attempted but yielded results far worse than the CNN classifier method, so it is not included in this study.

Two classifiers of this kind are trained, using a dataset of 400 000 samples, consisting an equal fraction of noise events and one of the two classes of signal (BBH and BNS) considered in this study. These classifiers used the same data split (64% train, 16% validation, 20% test) as the AE datasets. The training is performed minimizing a binary cross entropy error loss function on the training sample of section 3, using the validation set to optimize the training and the test set to evaluate the model performance.

The classifier is tested on noise samples, as well as on BBH and BNS events. When tested on the same kind of signal it is trained on, the classifier accuracy is used to estimate the best accuracy that the AE could reach and, consequently, the loss in accuracy due to the use of an unsupervised approach. When testing the classifier on the signal it was not trained on, we can instead compare the generalization property of the AE to that of a supervised algorithm. The two tests provide an assessment of the balance between accuracy and generalization power and demonstrate the complementarity between our approach and a standard template-based method. In practice, one could implement as many supervised algorithms as known GW sources, while using an unsupervised algorithm to be sensitive to unexpected signal sources (and

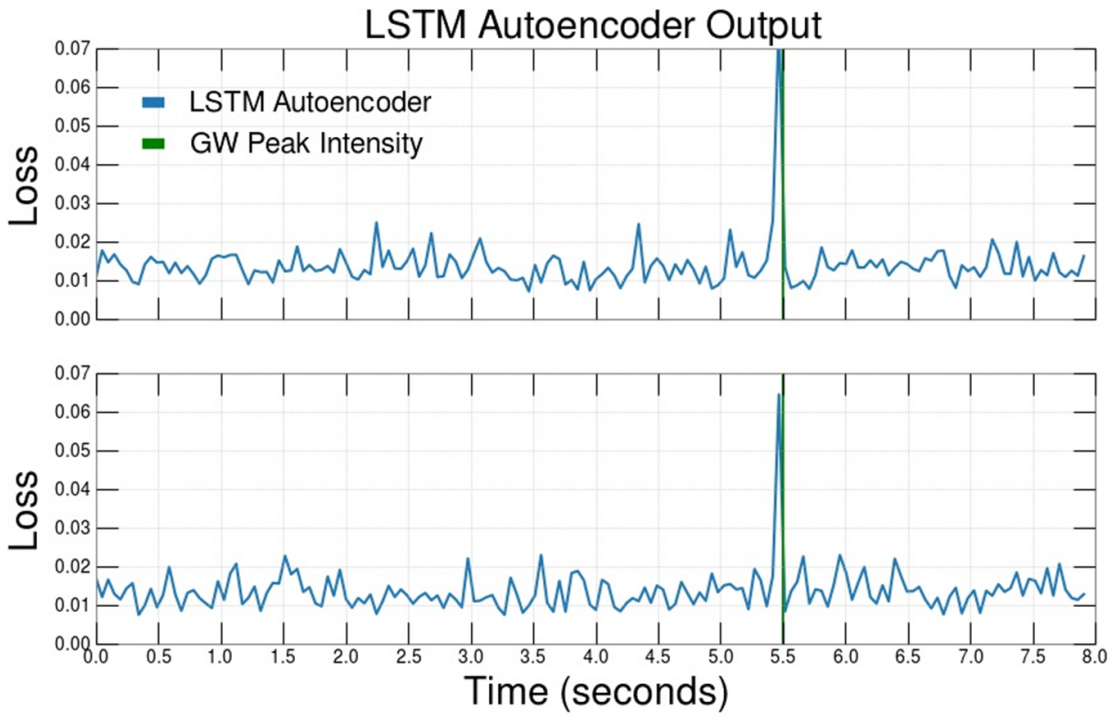

Figure 3. Spike/dip in MSE loss for an LSTM AE over one 8 s event sampled at 2048 Hz (split into 100-timestep inputs) signaling the detection of a GW for the H1 (top) and L1 (bottom) detectors at an SNR = 18.1. The AE models are never presented with any GW signals (green) during training. Detection thresholds are set by fixing a FPR, which is done continuously to create a ROC curve in section 5.

non-coincident signals across multiple interferometers, in cases of glitch detection and data quality monitoring).

5. Results

Figure 4 shows the receiver operating characteristic (ROC) curves for three AE architectures (LSTM, GRU, CNN). The curves are obtained considering a single detector, i.e. no coincidence is enforced at this stage. In the left (right) figure, the ROC curves are evaluated on noise and a signal sample of BBH (BNS) merger data. For comparison, the CNN classifiers trained on both datasets are shown.

As the ROC curves show, the LSTM architecture provides the best accuracy among the AEs. The LSTM AE accuracy is worse than that of the classifier trained on the correct signal hypothesis, but better than that of the classifier trained on the opposing signal hypothesis. Remarkably, the AE outperforms the classifier for a tight enough selection on the network score, i.e. for a low enough target false positive rate (FPR) value. The performance comparison is quantified in table 1, where the FPRs corresponding to fixed values of the true positive rates (TPRs) are shown. The ROC curves and the values shown in table 1 quantify the trade-off between accuracy and generalization that motivates this study. This makes AEs especially useful to potentially discover unexpected GW sources, as well as GW sources that cannot be modeled by traditional simulation techniques.

This advantage is especially marked with BBHs, which have larger masses and thus shorter signals for the same integrated SNR, leading to a higher instantaneous amplitude at the merger. Thus, AE architectures will likely have an advantage with higher mass-range mergers in the regime where supervised learning models cannot generalize. This is as opposed to BNSs, which have lower mass values and consequently lower amplitude and higher frequency signatures. In this case, the generalization performance stagnates for both of the models, meaning that both models are extracting the same amount of signal out of the events. Still, in the AUC metric the AE models performs equivalently to the supervised algorithm trained on the wrong signal hypothesis and over-performs at low FPR.

The TPR values quoted on table 1 are obtained averaging across the SNR, which is uniformly distributed in a [5, 20] range. On the other hand, the TPRs of the AE models depend strongly on the SNR value, as shown in figure 5 both for BBH and BNS merger events. As shown in the figure, the LSTM AE guarantees

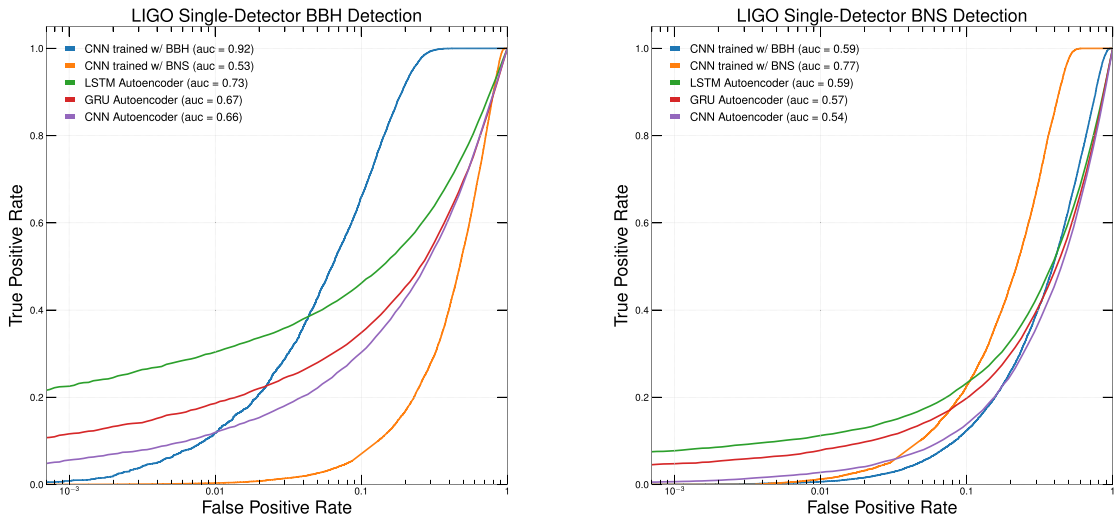

Figure 4. Single-detector ROC curves for the LSTM, GRU, CNN AEs compared to the corresponding CNN supervised architecture. The figure shows the ROC curves for supervised CNNs trained on the BBH and BNS merger data, when the inference is performed on noise and a signal sample of BBH (left) or BNS (right) events. 'Area under the ROC curve' (AUC) measures the two-dimensional area under a ROC curve and quantifies the overall performance of a model. An AUC = 0.50 corresponds to a classifier making random predictions.

Table 1. True-positive rates for BBH and BNS merger detection at 10% and 1% false-positive rates, for AEs trained on noise and for binary CNN classifiers trained on BBH and BNS simulations. The AE architecture with the best unsupervised results is marked in bold.

BBH signal vs. noise					
FPR	LSTM-AE	GRU-AE	CNN-AE	BBH-CNN	BNS-CNN
0.1	**46.1**%	34.8%	30.2%	66.0%	10.0%
0.01	**30.4**%	18.6%	12.0%	11.8%	0.3%
BNS signal vs. noise					
FPR	LSTM-AE	GRU-AE	CNN-AE	BBH-CNN	BNS-CNN
0.1	**23.7**%	20.2%	14.4%	12.4%	22.5%
0.01	**11.4**%	8.16%	2.9%	0.6%	1.6%

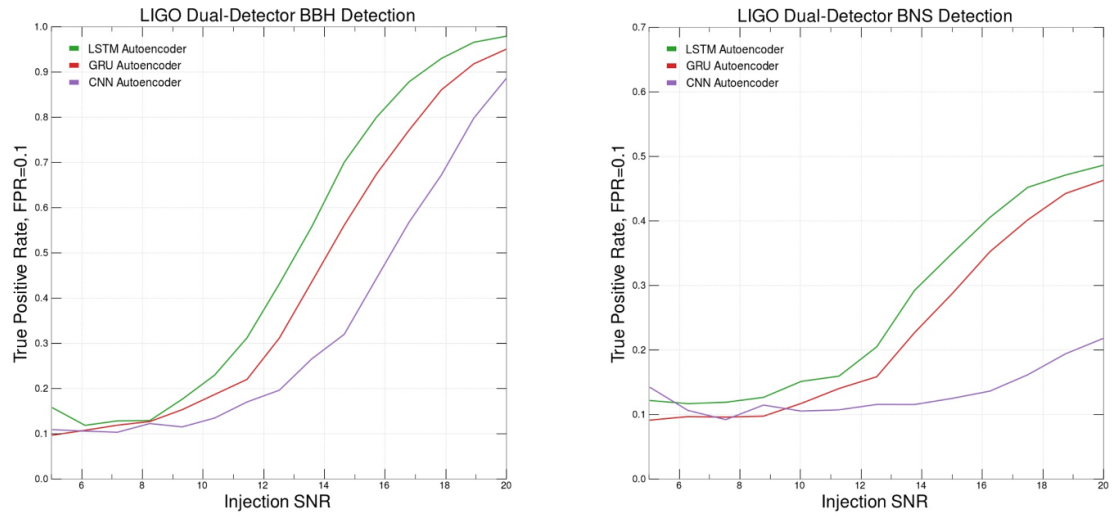

Figure 5. TPRs for the LSTM, GRU, CNN AEs at a fixed FPR (FPR = 0.1) with BBH (left) and BNS (right) events at variable SNRs.

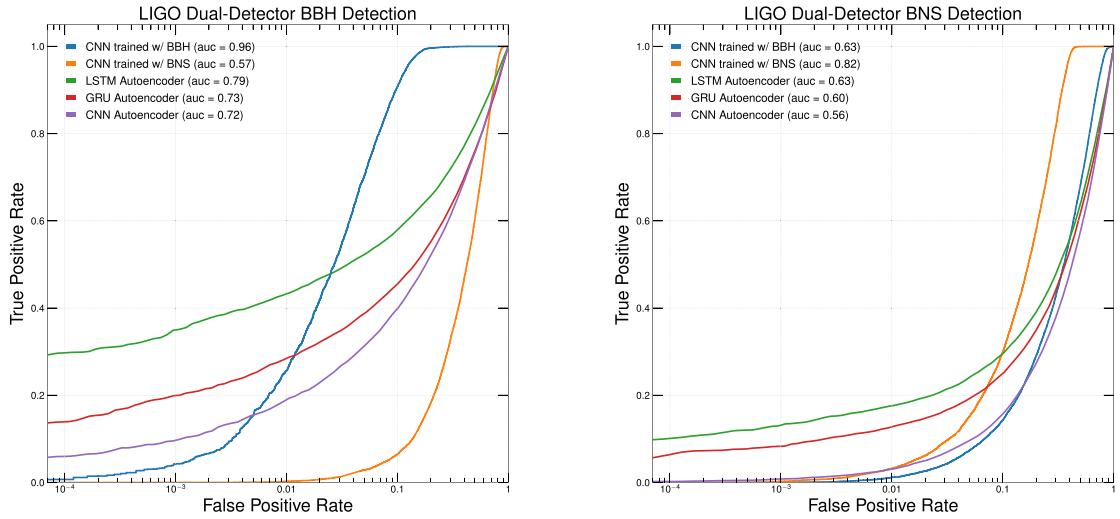

Figure 6. ROC curves for the LSTM, GRU, CNN AEs, obtained by exploiting signal coincidence on two detectors. The loss is computed independently on each detector and the same threshold is applied. The test is performed on noise and a signal sample of BBH (left) or BNS (right) events. 'Area under the ROC curve' (AUC) measures the two-dimensional area under a ROC curve and quantifies the overall performance of a model. An AUC = 0.50 corresponds to a classifier making random predictions.

better performance across the considered range of SNR values. While the improvement (e.g. with respect to the CNN AE) is roughly constant for BBH events, in the case of BNS events the LSTM AE is particularly better than the other architectures for large SNR values. Overall, this study reinforces the idea that the LSTM AE is the most robust choice among those we considered.

In a realistic exploitation of this algorithm, one would define a threshold above which the data would be called a potential signal. Doing so, one would like to keep the FPR at a manageable rate, while retaining a reasonable TPR value. For instance, an FPR of 10^{-4} would correspond to about one false alarm a day, low enough for a post detection assessment of the nature of the anomaly. Similarly, a FPR of 10^{-6} would correspond to about one false alarm every three months, low enough for the algorithm to be used in a real-time data processing, e.g. to serve as a trigger for multi-messenger astronomy.

The key ingredient to reach low FPR values is the exploitation of signal coincidence across multiple detectors. Since the noise across detectors is uncorrelated, single-detector FPR of $\sim 10^{-2}$ (as in table 1) would give a global FPR of $\sim 10^{-4}$ when two instruments are put in coincidence. Clearly, the presence of uncorrelated noise overlapped to the signal dilutes the correlation of the anomaly across different devices. On the other hand, a certain level of correlation is retained. For instance, we observe a 40% correlation on the LSTM anomaly score for BBH merger events. For comparison, a 70% correlation is observed for the CNN classifier. Coincidence can be enforced requiring that two signals above a certain threshold are detected at the same time. Alternatively, one could apply a threshold on the sum of the two losses, with the idea that an MSE loss function is loosely related to the negative log likelihood, so that the sum of the loss would correspond to the negative log of the likelihood products. The former approach has the advantage of requiring the two detectors to communicate only after the anomaly event in a detector is identified. This means that the data throughput to be transmitted can be kept low. On the other hand, the latter approach provides better performance and it is considered here. In this case, one would have to find solutions to mitigate the data throughput and facilitate the communication of the detectors in real time. For instance, one could run the encoder at each experiment site and transmit the compressed data, with the decoding and coincidence check happening off-site. One should keep in mind that the LSTM model can run on a Field Programmable Gate Array within $\mathcal{O}(100)$ nsec, as demonstrated in [54].

To show this, we consider the case of two detectors and we build a ROC curve requiring a signal above a threshold on the sum of the AE losses. The result is shown in figure 6, both for BBH and BNS mergers and quantified in table 2. Keeping as a target a FPR of 10^{-4}, one can retain a BBH TPR comparable to that of the 10^{-2} FPR of the single-detector threshold (see table 1), while reducing the FPR by two orders of magnitude. The situation is qualitatively similar for BNS, with some quantitative difference: the two-detector combination comes with a $\sim 20\%$ ($\sim 10\%$) relative reduction of the TPR for a FPR of 10^{-2} (10^{-4}). Even taking into account, this efficiency loss, there is still a striking advantage in exploiting the coincidence of the signal across detectors to suppress the noise.

Table 2. True-positive rates for BBH and BNS merger detection at single-detector 10^{-1} and 10^{-2} false-positive rates corresponding to dual-detector FPRs of 10^{-2} and 10^{-4}, obtained by exploiting signal coincidence in two detectors. The AE architecture with the best unsupervised results is marked in bold.

BBH signal vs. noise					
FPR	LSTM-AE	GRU-AE	CNN-AE	BBH-CNN	BNS-CNN
0.01	**43.3**%	28.5%	18.9%	25.6%	0.2%
0.0001	**29.7**%	13.9%	6.0%	0.7%	0.0%
BNS signal vs. noise					
FPR	LSTM-AE	GRU-AE	CNN-AE	BBH-CNN	BNS-CNN
0.01	**17.6**%	12.7%	3.1%	1.1%	3.2%
0.0001	**10.1**%	6.3%	0.2%	0.0%	0.0%

6. Conclusions

We presented an unsupervised strategy to detect GW signals from unspecified sources exploiting an AE trained on noise. The AE is trained to compress input data to a low-dimension latent space and reconstruct a representation of the input from the point in the latent space. The algorithm is optimized using as a loss function a differentiable metric, quantifying the distance between input and output data. Given a trained AE, one could identify anomalous data isolating the outlier data populating the tail of the loss distribution.

We applied this strategy to a sample of synthetic data from two GW interferometers. We explore different choices for the network architecture and compare the single-detector detection capability to that of a CNN binary classifier, trained on specific signal hypotheses. We show how a recurrent AE provides the best anomaly detection performance on benchmark BBH and BNS merger events. We show the trade-off between accuracy and generalization, when using this unsupervised strategy rather than a supervised approach (here quantified through a CNN binary classifier trained on the same data and the corresponding labels). Additionally, we show that unsupervised AEs perform better than supervised classifiers (trained on the same amount of data) at low enough target FPR values.

We show how the coincidence of two detectors, both selecting anomalies at an expected FPR $\sim 10^{-2}$, would retain a TPR of 43.3% (17.6%) for BBH (BNS) signals while giving one false alarm a day which can be easily be discarded after a post-detection analysis, e.g. with more traditional GW detection strategies. By comparing this performance to that achieved by other network architectures, we show that our network design, and in particular the use of LSTMs to exploit the time-series nature of the data, represents a progress in terms of anomaly detection capabilities.

With the same FPR, one could bring the false alarm rate to about once every three month, exploiting the coincidence of three detectors. In this case, the algorithms proposed here could operated in the real-time as part of a trigger system for multi-messenger astronomy, in the spirit of what is discussed in [55] for real-time data analysis at the Large Hadron Collider. Considering the relatively low computational cost of such an algorithm [54] and the high impact of a potential signal detection by this algorithm, its implementation for LIGO and VIRGO would be certainly beneficial, despite the fact that the expected detection probability cannot be guaranteed to be high for any signal source.

Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI: 10.5281/zenodo.5121514, 10.5281/zenodo.5121510, 10.5281/zenodo.5772814 and 10.5281/zenodo.5773513.

Acknowledgments

We are grateful to the insight and expertise of Rana Adhikari, Hang Yu, and Erik Katsavounidis from the LIGO collaboration and Elena Cuoco from the VIRGO collaboration, who guided us on a field of research which is not our own.

Part of this work was conducted at '*iBanks*', the AI GPU cluster at Caltech. We acknowledge NVIDIA, SuperMicro and the Kavli Foundation for their support of '*iBanks*'.

This work was carried on as part of the 2020 CERN OpenLab Summer Student program, which was carried on in remote mode due to the COVID pandemic.

M P is supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (Grant Agreement No. 772369).

E M is supported by the Institute for Research and Innovation in Software for High Energy Physics (IRIS-HEP) through a fellowship in Innovative Algorithms.

This work is partially supported by the U.S. DOE, Office of Science, Office of High Energy Physics under Award Nos. DE-SC0011925, DE-SC0019227 and DE-AC02-07CH11359.

ORCID iDs

Eric A Moreno https://orcid.org/0000-0001-5666-3637
Bartlomiej Borzyszkowski https://orcid.org/0000-0002-2927-7009
Maurizio Pierini https://orcid.org/0000-0003-1939-4268
Jean-Roch Vlimant https://orcid.org/0000-0002-9705-101X
Maria Spiropulu https://orcid.org/0000-0001-8172-7081

References

[1] Aasi J *et al* 2015 Advanced LIGO *Class. Quantum Grav.* **32** 074001
[2] Acernese F *et al* 2015 Advanced Virgo: a second-generation interferometric gravitational wave detector *Class. Quantum Grav.* **32** 024001
[3] Abbott B P *et al* 2016 Observation of gravitational waves from a binary black hole merger *Phys. Rev. Lett.* **116** 061102
[4] Abbott B P *et al* 2017 Erratum: GW170104: observation of a 50-solar-mass binary black hole coalescence at redshift 0.2 [2017 *Phys. Rev. Lett.* **118** 221101] *Phys. Rev. Lett.* **121** 129901
[5] Abbott B P *et al* 2017 GW170817: observation of gravitational waves from a binary neutron star inspiral *Phys. Rev. Lett.* **119** 161101
[6] Abbott B P *et al* 2017 Multi-messenger observations of a binary neutron star merger *Astrophys. J. Lett.* **848** L12
[7] Allen B *et al* 2012 FINDCHIRP: an algorithm for detection of gravitational waves from inspiraling compact binaries *Phys. Rev.* D **85** 122006
[8] Sathyaprakash B S and Dhurandhar S V 1991 Choice of filters for the detection of gravitational waves from coalescing binaries *Phys. Rev.* D **44** 3819–34
[9] Balasubramanian R, Sathyaprakash B S and Dhurandhar S V 1996 Gravitational waves from coalescing binaries: detection strategies and monte carlo estimation of parameters *Phys. Rev.* D **53** 3033–55
[10] Owen B J 1996 Search templates for gravitational waves from inspiraling binaries: choice of template spacing *Phys. Rev.* D **53** 6749–61
[11] Cokelaer T 2007 Gravitational waves from inspiralling compact binaries: hexagonal template placement and its efficiency in detecting physical signals *Phys. Rev.* D **76** 102004
[12] Smith R *et al* 2016 Fast and accurate inference on gravitational waves from precessing compact binaries *Phys. Rev.* D **94** 044031
[13] Klimenko S *et al* 2016 Method for detection and reconstruction of gravitational wave transients with networks of advanced detectors *Phys. Rev.* D **93** 042004
[14] Huerta E A *et al* 2017 Complete waveform model for compact binaries on eccentric orbits *Phys. Rev.* D **95** 024038
[15] Huerta E A and Brown D A 2013 Effect of eccentricity on binary neutron star searches in advanced LIGO *Phys. Rev.* D **87** 127501
[16] Huerta E A *et al* 2014 Accurate and efficient waveforms for compact binaries on eccentric orbits *Phys. Rev.* D **90** 084016
[17] Drago M *et al* 2021 Coherent WaveBurst, a pipeline for unmodeled gravitational-wave data analysis *SoftwareX* **14** 100678
[18] Baker P T, Sarah Caudill K A Hodge D T, Capano C and Cornish N J 2015 Multivariate classification with random forests for gravitational wave searches of black hole binary coalescence *Phys. Rev.* D **91** 062004
[19] George D and Huerta E A 2018 Deep neural networks to enable real-time multimessenger astrophysics *Phys. Rev.* D **97** 044039
[20] George D and Huerta E A 2018 Deep learning for real-time gravitational wave detection and parameter estimation: results with advanced LIGO data *Phys. Lett.* B **778** 64–70
[21] Gabbard H *et al* 2018 Matching matched filtering with deep networks for gravitational-wave astronomy *Phys. Rev. Lett.* **120** 141103
[22] Jadhav S, Mukund N, Gadre B, Mitra S and Abraham S 2021 Improving significance of binary black hole mergers in advanced LIGO data using deep learning: confirmation of GW151216 *Phys. Rev.* D **104** 064051
[23] Fukushima K 1980 Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position *Biol. Cybern.* **36** 193–202
[24] LeCun Y *et al* 1999 Object recognition with gradient-based learning *Shape, Contour and Grouping in Computer Vision* (Berlin: Springer) pp 319–45
[25] McGinn J, Messenger C, Heng I S and Williams M J 2021 Generalised gravitational wave burst generation with generative adversarial networks *Class. Quantum Grav.* **38** 155005
[26] Colgan R E *et al* 2020 Efficient gravitational-wave glitch identification from environmental data through machine learning *Phys. Rev.* D **101** 102003
[27] Rumelhart D E, Hinton G E and Williams R J 1986 *Learning Internal Representations by Error Propagation* (Cambridge, MA: MIT Press) pp 318–62
[28] Morawski F *et al* 2021 Anomaly detection in gravitational waves data using convolutional autoencoders *Mach. Learn.: Sci. Technol.* **2** 045014
[29] Hochreiter S and Schmidhuber J 1997 Long short-term memory *Neural Comput.* **9** 1735–80
[30] Chung J *et al* 2014 Empirical evaluation of gated recurrent neural networks on sequence modeling *CoRR* (arXiv:1412.3555)
[31] Kapadia S J, Dent T and Canton T D 2017 Classifier for gravitational-wave inspiral signals in nonideal single-detector data *Phys. Rev.* D **96** 104015
[32] Miller A L *et al* 2019 How effective is machine learning to detect long transient gravitational waves from neutron stars in a real search? *Phys. Rev.* D **100** 062005
[33] Huerta E A *et al* 2021 Accelerated, scalable and reproducible AI-driven gravitational wave detection *Nat. Astron.* **5** 1062–8

[34] Veitch J *et al* 2015 Parameter estimation for compact binaries with ground-based gravitational-wave observations using the LALInference software library *Phys. Rev.* D **91** 042003
[35] Ashton G *et al* 2019 Bilby: a user-friendly Bayesian inference library for gravitational-wave astronomy *Astrophys. J. Suppl. Ser.* **241** 27
[36] Romero-Shaw I M *et al* 2020 Bayesian inference for compact binary coalescences with BILBY: validation and application to the first LIGO–Virgo gravitational-wave transient catalogue *Mon. Not. R. Astron. Soc.* **499** 3295–319
[37] Benkő Z, Bábel T and Somogyvári Z 2020 How to find a unicorn: a novel model-free, unsupervised anomaly detection method for time series (arXiv:2004.11468 [cs.LG])
[38] Karl Pearson F R S 1901 LIII. On lines and planes of closest fit to systems of points in space *London, Edinburgh Dublin Phil. Mag. J. Sci.* **2** 559–72
[39] Powell J *et al* 2015 Classification methods for noise transients in advanced gravitational-wave detectors *Class. Quantum Grav.* **32** 215012
[40] Powell J, Torres-Forné A, Lynch R, Trifirò D, Cuoco E, Cavaglià M, Heng I S and Font J A 2017 Classification methods for noise transients in advanced gravitational-wave detectors II: performance tests on advanced LIGO data *Class. Quantum Grav.* **34** 034002
[41] Philip N S and Joseph K B 2000 Boosting the differences: a fast Bayesian classifier neural network *Intell. Data Anal.* **4** 463–73
[42] Mukund N *et al* 2017 Transient classification in LIGO data using difference boosting neural network *Phys. Rev.* D **95** 104059
[43] Gebhard T and Kilbertus N 2019 Generate gravitational-wave data (GGWD) (available at: https://github.com/timothygebhard/ggwd)
[44] Moreno E 2021 Source-agnostic gravitational-wave detection with recurrent autoencoders: BBH dataset *Zenodo* (available at: https://doi.org/10.5281/zenodo.5772814)
[45] Moreno E 2021 Source-agnostic gravitational-wave detection with recurrent autoencoders: BNS dataset *Zenodo* (available at: https://doi.org/10.5281/zenodo.5773513)
[46] LIGO Scientific Collaboration 2018 LIGO algorithm library—LALSuite. Free software (GPL)
[47] Nitz A *et al* 2020 gwastro/pycbc: PyCBC release v1.16.9 *Zenodo* (available at: https://doi.org/10.5281/zenodo.3993665)
[48] Rich A, Rana A, Stefan B, Lisa B, Matt E, Peter F, Valera F, Guido M, Bram S and Sam W (LIGO Scientific Collaboration) 2008 AdvLIGO Interferometer Sensing and Control Conceptual Design
[49] Bohé A *et al* 2017 Improved effective-one-body model of spinning, nonprecessing binary black holes for the era of gravitational-wave astrophysics with advanced detectors *Phys. Rev.* D **95** 044028
[50] Abbott B P *et al* 2019 Properties of the binary neutron star merger GW170817 *Phys. Rev.* X **9** 011001
[51] Özel F, Psaltis D, Narayan R and Villarreal A S 2012 On the mass distribution and birth masses of neutron stars *Astrophys. J.* **757** 55
[52] Mandel I and O'Shaughnessy R 2010 Compact binary coalescences in the band of ground-based gravitational-wave detectors *Class. Quantum Grav.* **27** 114007
[53] Cuoco E *et al* 2001 On line power spectra identification and whitening for the noise in interferometric gravitational wave detectors *Class. Quantum Grav.* **18** 1727–52
[54] Que Z *et al* 2021 Accelerating recurrent neural networks for gravitational wave experiments (arXiv:2106.14089)
[55] Cerri O *et al* 2019 Variational autoencoders for new physics mining at the Large Hadron Collider *J. High Energy Phys.* **05** 036