Ensemble of deep convolutional neural networks for real-time gravitational wave signal recognition

CunLiang Ma,[1] Wei Wang,[1] He Wang[ORCID],[2] and Zhoujian Cao[ORCID][3,4,*]

[1]*School of Information Engineering, Jiangxi University of Science and Technology, Ganzhou, 341000, China*
[2]*CAS Key Laboratory of Theoretical Physics, Institute of Theoretical Physics, Chinese Academy of Sciences, Beijing 100190, China*
[3]*Department of Astronomy, Beijing Normal University, Beijing 100875, China*
[4]*School of Fundamental Physics and Mathematical Sciences, Hangzhou Institute for Advanced Study, UCAS, Hangzhou 310024, China*

With the rapid development of deep learning technology, more and more researchers apply it to gravitational wave (GW) data analysis. Previous studies focused on a single deep learning model. In this paper we design an ensemble algorithm combining a set of convolutional neural networks for GW signal recognition. The whole ensemble model consists of two subensemble models. Each subensemble model is also an ensemble model of deep learning. The two subensemble models treat data of Hanford and Livinston detectors, respectively. Proper voting scheme is adopted to combine the two subensemble models to form the whole ensemble model. We apply this ensemble model to all reported GW events in the first observation and second observation runs (O1/O2) by LIGO-VIRGO Scientific Collaboration. We find that the ensemble algorithm can clearly identify all binary black hole merger events except GW170818. We also apply the ensemble model to one month (August 2017) data of O2. No false trigger happens, although only O1 data are used for training. Our test results indicate that the ensemble learning algorithms can be used in real-time GW data analysis.

I. INTRODUCTION

On September 14, 2015, the advanced Laser Interferometer Gravitational Wave Observatory (aLIGO) directly detected a gravitational wave (GW) event, GW150914, for the first time in human history [1]. This successful detection directly verified the prediction of the existence of GW in general relativity and opened a new window of universe observation. Such detection opens the era of gravitational wave astronomy. Gravitational wave astronomy is important to the study of the origin and evolution of the universe, the nature of dark matter and dark energy, and others. On August 17, 2017, the gravitational wave event GW170817 [2] produced by the merging of a binary neutron star was directly observed. This was the first time that GW, gamma-ray burst, optics, and other electromagnetic signals [3–5] were clearly and continuously detected from one astrophysical source [6]. The observations of GW170817 marked the arrival of the era of multimessenger astronomy including GW.

Gravitational wave astronomy requires multidisciplinary collaborative research, such as mathematics, physics, and computer science. Since the detected gravitational wave signal is buried in a background noise, elaborate data analysis is needed to extract the GW signal. The widely used data analysis method is matched filtering. With matched filtering, each data segment in the detected strain is matched to a template bank. The maximum value of the output is the matched signal-to-noise ratio (SNR). When it is bigger than a specified threshold, a GW signal candidate is announced. Subsequently, further inspection methods such as χ^2 time-frequency test [7], multidetector coincidence test [8], and others are used to calculate the false-alarm rate of the candidate which is used to determine whether it is a true GW event or not. Since the parameters of the potential GW source are not known in advance, the parameter space of the template bank must be widely searched. The matched-filtering method is computationally expensive. The low-latency implementation of the matched-filtering method cannot be extended to the nine-dimensional signal manifold [9]. Recently, research has shown that the method based on the artificial neural network may replace matched filtering to achieve efficient detection.

As early as the 1980s, many researchers were engaged in the study of neural networks [10–13]. However, the application range of neural networks was very limited at that time, it was only used to deal with small-size picture

[*]zjcao@amt.ac.cn

recognition (such as handwritten digital recognition [14]). Because of the impact of the vanishing gradient problem [15], the neural network cannot improve the accuracy through deepening the model structure. This problem seriously hindered the development of the neural network. The situation changed in 2012. The AlexNet won the championship in the ImageNet competition based on the deep Convolutional Neural Network (CNN) [16]. At present, the CNN has achieved great success in many fields such as image classification [17–19], image segmentation [20–22], object detection [23–25], natural language processing, and speech recognition [26–28].

In 2018, George *et al.* [29] and Gabbard *et al.* [30] individually concluded that deep learning can achieve the accuracy as matched filtering in gravitational wave data analysis. Compared to matched filtering, the computational efficiency of deep learning is much higher. Deep learning is expected to be applied to real-time detection of transient GW. Recently, the application of the CNN in GW research has been expanded greatly [31–38].

Figure 1 shows the comparison of CNN and matched filtering. To facilitate the comparative analysis, we give a brief mathematical description of the matched filtering. The matched output corresponding to the *ith* template in template bank is

$$z_i[n] = T\sum_m x[m]h_i[m - n], \quad (1)$$

where T is sampling period, m and n represent sampling time, $x[n]$ is the whitened strain, and $h_i[n]$ is the *i*th template whitened by background noise. Equation (1) indicates that $z_i[n]$ is convolutional results between $x[n]$ and $h_i[-n]$. The matched SNR of the strain can be calculated by

$$z[n] = \max_i(z_i[n]). \quad (2)$$

From the above analysis, we can get a conclusion that matched filtering can be understood as a convolutional neural network with only one convolutional layer composed of a large number of kernels comprising the template bank and this conclusion agrees with work of Jingkai Yan *et al.* [39]. The deep learning model effectively compresses a large number of approximated template waveforms from the dimension of width to the dimension of depth. Some recent research shows that the deep learning model has a relatively high probability of misjudging the noise as a GW signal [40]. The ensemble of multiple deep learning models may simultaneously focus on the width and depth of the approximated template waveforms and is expected to improve the effect of gravitational wave detection.

Ensemble learning can improve the performance of the learner via a combination of multiple learners. In 1979, Dasarathy *et al.* [41] first proposed the concept of an ensemble system, which opened the prelude to the
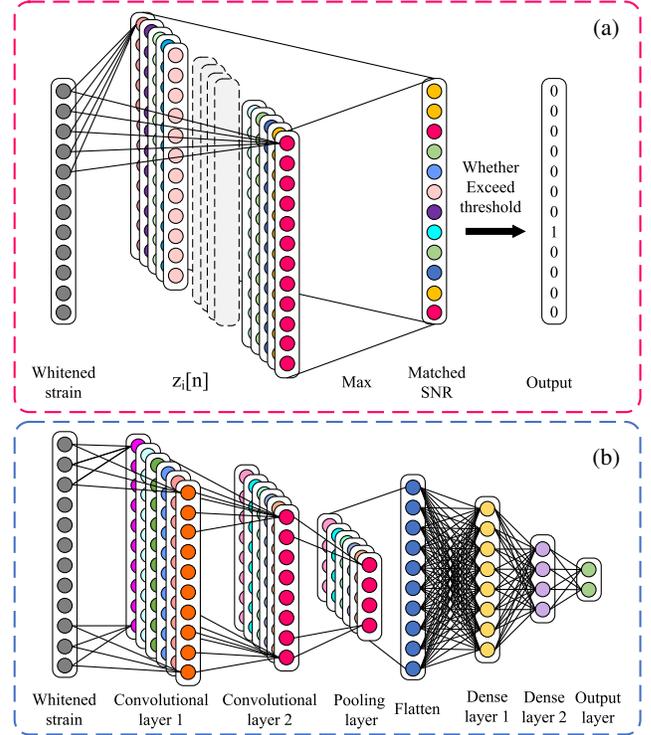
FIG. 1. Diagram of matched filtering (a) and CNN (b). The channel with gray circles represents the whitened strain. In second layer, the various channels with colored circles are the convolutional output of different kernels and the circles in a channel represent the output at different times. Note that the channels of second layer in (a) are not the templates. In (a), the convolutional kernels of second layer are templates whitened by the background noise while in (b) the kernels are gotten via training.

development of ensemble learning. In 1990, Hansen *et al.* [42] proved that the ensemble system has the characteristic of variance reduction. The discovery of this property proves that ensemble learning has the ability to improve the effect of neural networks and provides a theoretical basis for neural networks combined with ensemble learning. In 2021, Huerta *et al.* [43] designed a model for GW detection which is composed of two optimized WaveNet. Their model realized real-time signal recognition. Recently, they developed a GW detection workflow with an artificial intelligence (AI) ensemble. Using this AI ensemble, they only spent 7 min to analyze all LIGO data in 2017 August. All signals are found without false alarm [44]. In the current work, we build an ensemble model to detect real GW events by using the optimized bagging [45] algorithm and successfully detect almost all GW events in O1 and O2. The most obvious difference between our method and the method in [44] is that we individually build two ensemble models for Hanford and Livingston interferometers, respectively. Differently, Huerta *et al.* [44] use feature fusion to combine the strain data of two interferometers for each

model in the ensemble. The individual model of the ensemble is also called base learner.

The χ^2 time-frequency test together with the matched-filtering method can eliminate the false trigger caused by a glitch. However, χ^2 time-frequency test cannot be used together with deep learning. We use the strategy of cross validation of two detectors to treat this problem. Since the probability of the simultaneous presence of glitch in the detected noise signal from two interferometers is extremely low, combining the detection results from the two interferometers can effectively reduce the number of false triggers caused by the glitch. This is the basic idea for the current work. The test results based on August 2017 LIGO data indicates that our idea works very well.

This paper is organized as follows. In Sec. II, we introduce the process of building data for training and testing. After that we describe our ensemble model in Sec. III. Then we apply our ensemble model to the O1/O2 LIGO data in Sec. IV. The high true-alarm rate and extremely low false-alarm rate behavior of our ensemble model will be reported there. The last section is devoted to a summary.

II. DATA FOR TRAINING AND TESTING

The dataset for the deep model in this work was generated by the open-source project ggwd[1] [46]. We focus on gravitational wave signals produced by binary black hole (BBH) mergers. Two classes of data are generated. One is the data strain of pure background noise and the other is the data strain including background noise and a GW waveform. The GW waveform contains the complete inspiral, merger, and ringdown phases of the BBH coalescence. The synthetic data can be represented by

$$s(t) = h(t) + n(t), \tag{3}$$

where $h(t)$ represents the GW waveform which is generated by the effective one-body numerical relativity waveform model [47–50]. $n(t)$ is the background noise that is randomly sampled from the O1 data and all of the identified GW events are eliminated.

To filter out the spectral components of the environmental noise and eliminate the influence of Newtonian noise, all the data in the dataset were whitened and passed through a high-pass filter with a cutoff frequency of 20 Hz. The duration of the training dataset is 1 s and the sampling rate is 4096 Hz.

The mass of both black holes is randomly sampled between 5 $M_\odot$ and 80 $M_\odot$, and the dimensionless spin is randomly sampled in (0, 0.998). The polarization angle is sampled uniformly at random from the interval $(0, 2\pi)$. The coalescence phase and the inclination angle are sampled

jointly from a uniform distribution over a sphere, while the right ascension and the declination are all sampled from a uniform distribution over the sky. The distance between the Earth and the source is a fixed value of 100 Mpc. But, the simulated waveforms are later rescaled to match a given network SNR [51] which is randomly sampled in (7,20).

A dataset of 120,000 samples was generated, within 45,000 samples containing GW signal and 75,000 samples containing pure background noise. We use 60,000 samples (30,000 GW and 30,000 noise) for training, 10,000 samples (5,000 GW and 5,000 noise) for validation, and 50,000 samples (10,000 GW and 40,000 noise) for testing. In order to mimic the fact that the number of noise samples in actual gravitational wave detection are much larger than that of the gravitational wave samples, we did not adopt the conventional scheme which generate a test set with the 50% noise samples and 50% GW samples. Alternatively, we make the test set with 80% noise samples and 20% GW samples.

III. ENSEMBLE DEEP LEARNING SCHEME FOR GRAVITATIONAL WAVE DATA ANALYSIS

This section exhibits the proposed ensemble model for GW data analysis. We apply the bagging algorithm [45] to build the ensemble model. Firstly, we introduce the structure of the base learners. Secondly, we present the proposed detection method consisting of two independent ensemble models, and each model detects the data of Hanford and Livingston interferometer, respectively. Then, the training method of the base learners and the method for the model selection will be detailed.

A. Base learner

In recent years, many works investigated the application of the CNN to the GW detection [29,52–55]. The CNN here we used contains one input layer, several hidden layers, and one output layer to form the base learners of the ensemble deep learning model.

Considering the good performance of the model designed by Gabbard *et al.* [30], we adopt this model as the base learner of our ensemble model. In the original work [30], the simulated Gaussian noise was used. After using the optimized ensemble method and real background noise, an ensemble model with better detection capability can be constructed. We find that the ensemble model works well for real background noise and surmise that the dropout [56] augments the diversity of the base model to make it suitable to be assembled.

The public data recorded by LIGO include two types which correspond to sampling rate 4096 and 16384 Hz. In the current work we treat the 4096-Hz one. The model designed in [30] is only applicable to data with the sampling rate 8192 Hz; we make some modifications to the structure of their model. Our base learner has six

---

[1]https://github.com/timothygebhard/ggwd

TABLE I.   The basic structure of base learner.

Layer	Input	Vector (size: 4096)
1	Reshape	Matrix (size: 1×4096)
2	Convolution	Matrix (size: 8×4033)
3	Convolution	Matrix (size: 8×4002)
4	Max pooling	Matrix (size: 8×500)
5	Convolution	Matrix (size: 16×469)
6	Convolution	Matrix (size: 16×454)
7	Max pooling	Matrix (size: 16×75)
8	Convolution	Matrix (size: 32×60)
9	Convolution	Matrix (size: 32×45)
10	Max pooling	Matrix (size: 32×11)
11	Flatten	Vector (size: 352)
12	Dense layer	Vector (size: 64)
13	Dropout	Vector (size: 64)
14	Dense layer	Vector (size: 64)
15	Dropout	Vector (size: 64)
	Output	Vector (size: 2)

convolutional layers, three max-pooling layers, and two dense layers, which are shown in Table I. The nonlinear activation function Elu is used after every layer except the last dense layer. The softmax activate function is added after the last dense layer. The output of the base learner returns a value in the range [0, 1] which indicates whether a BBH coalescence signal is present or not. To further enhance the diversity of the base learners, bootstrap and a data augmentation method are used. We use adaptive moment estimation with incorporated Nesterov momentum [57] with a momentum schedule of 0.004. The learning rate is set to 0.002, and the batch number value is 32.

B. Ensemble model

The architecture of the whole ensemble deep learning model is shown in Fig. 2. This ensemble model consists of two individual subensemble models which are used for Hanford and Livingston interferometers, respectively. Each subensemble model includes ten base learners. These ten base learners are chosen from 100 trained weak learners. The model training and selection methods will be described in the next section.

Here we describe the voting strategy used in the ensemble model. Both hard voting and soft voting are widely used in machine learning. Hard voting uses multiple individual models to make its predictions. Soft voting relies on probabilistic outcome values generated by classification algorithms. Compared to hard voting, soft voting takes into account more information. Based on this consideration our ensemble model adopts the soft voting scheme.

The average voting method can reduce the bias caused by abnormal voting results to a certain extent. In ensemble learning, average voting method is often used. We investigated two types of average voting schemes including arithmetic mean and harmonic mean. The detection

capability of the two methods will be shown in the following subsection.

We denote the output of the ith base learner in the ensemble model belonging to the interferometer X (Hanford or Livingston) as $y_{i,X}$, and the ensemble output as y_X. Then for the arithmetic mean, the average can be expressed as

$$y_X = \frac{1}{N} \sum_{i=1}^{N} y_{i,X}. \tag{4}$$

For the harmonic mean, the average can be expressed as

$$y_X = \frac{N}{\sum_{i=1}^{N} \frac{1}{y_{i,X}}}. \tag{5}$$

In the current work $N = 10$ is the number of the base learners of each detector. The detail of the model combination is introduced in the following subsection.

C. Model optimization

We use several skills to optimize our ensemble model. Some are about the training data. Some are about model construction. And, others are about model parameter setting. The following sub-subsections are for these optimizations.

1. Augmenting training data

In order to generate training data, we firstly use ggwd [46] to produce 60,000 samples (30,000 GW and 30,000 noise). Each sample data contain 8-s-long strain. When we inject a simulated GW signal into the strain, we always let the merger time of the BBH coalescence waveform locate at a fixed time. Specifically the merge occurs at 5.5 s in all GW strains. We named these data as original training set. Similarly, we generate 10,000 samples (5000 GW and 5000 noise) for validation and we named these data as original validation set. When we generate the noise for the training and validation data, only O1 data of LIGO are used. Although the training data are quite limited in this noise respect, the performance of our ensemble model is quite good as we will show in the following. This fact also indicates the robustness of our ensemble model for GW data analysis.

For training, We use bootstrap [58] to generate 100 sample data sets from the original training set. The detail of the bootstrap operation is as following. Firstly, we divide the 60,000 samples into 600 groups. Each group contains 100 samples. Then we randomly select group from these 600 groups. After each selection, the selected group will be put back and we randomly select again. Overall we do 600 selections. After these 600 selections we get a dataset which contains 60,000 samples. We repeat the above selection 100 times. Then we get 100 data sets. Each set contains 60,000 samples.
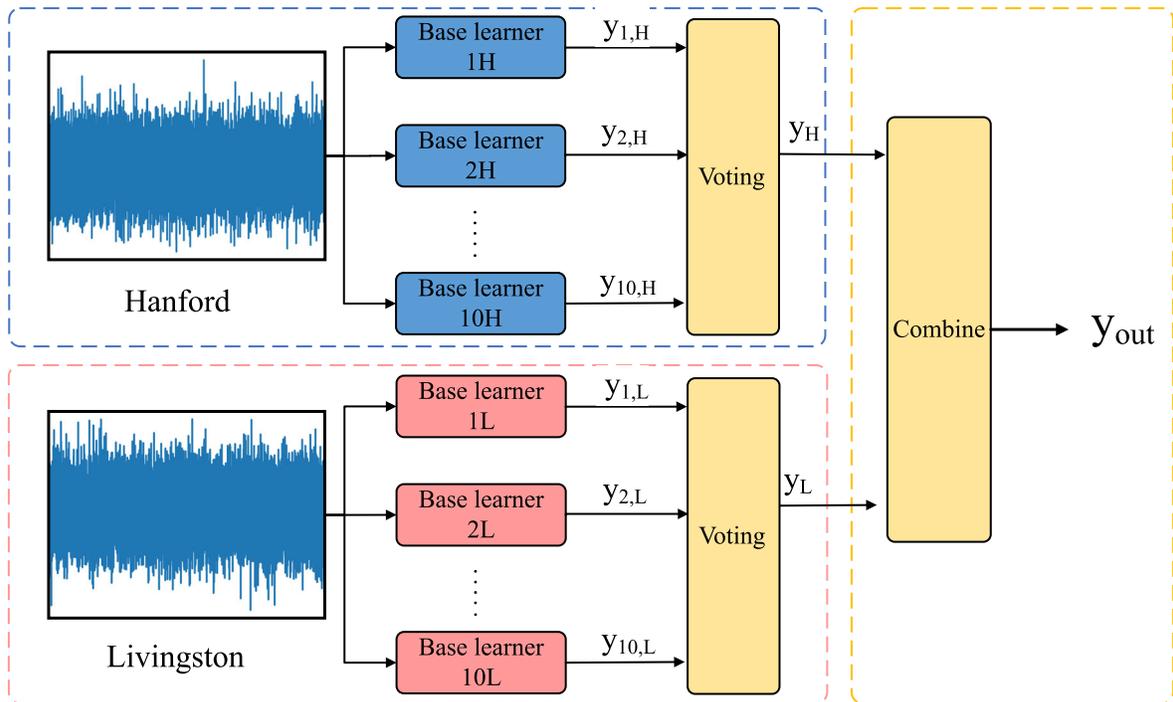
FIG. 2.   The structure of the ensemble deep learning model designed in the current work. Base learner iH or iL ($i \in [1, 10]$) denotes the ith base learner for the Hanford or Livingston detector, respectively. $y_{i,H}$ and $y_{i,L}$ represent the output of base learner iH and iL, respectively. The output y_{out} of the whole model is combined by the outputs y_H on Hanford detector and y_L on Livingston detector.

Now we have $100 \times 60,000$ data samples. Each sample contain 8-s-long strain. Based on each sample, we randomly extract two segments with duration 1 s and let the merger time of the BBH waveform locate in the range $\left(\frac{1}{8}, \frac{7}{8}\right)$ s. In Fig. 3 we show two examples of such extraction.

For validation, we adopt the same method as above but only randomly extract the data of the original validation set once to generate a validation set containing 10,000 1-s-long strain.

Through the above-described augmentation, we get 100 different training sets (each training set contain 120,000 data samples) for training and one validation set containing 10,000 data samples. Each sample contain 1-s strain. And, the merger time of the BBH waveform happens randomly between $\frac{1}{8}$ s and $\frac{7}{8}$ s.

2. Base learner construction

For each detector including Hanford or Livingston, we train 100 models individually. Since these models only aim to work for the specific training data, the generalization property may not be good. Consequently these models are traditionally called weak learner.

In machine learning, there are many ways to evaluate the performance of a model. For binary classification problems, the receiver operator characteristic (ROC) curves [59] can effectively reflect the model's performance. To draw the ROC curve, we need to calculate the true-alarm probability (TAP) and false-alarm probability (FAP) of

the model. We assume that in the test set, the subset containing GW signals is S_{GW}, and the subset only containing background noise is S_{N}. In the prediction results, the subset of samples predicted to be GW signal is S_{pGW}, and the subset of samples predicted to be noise is S_{pN}; then, TAP and FAP can be calculated according to the following formula:

$$\text{TAP} = \frac{S_{\text{GW}} \cap S_{\text{pGW}}}{S_{\text{GW}}}, \tag{6}$$
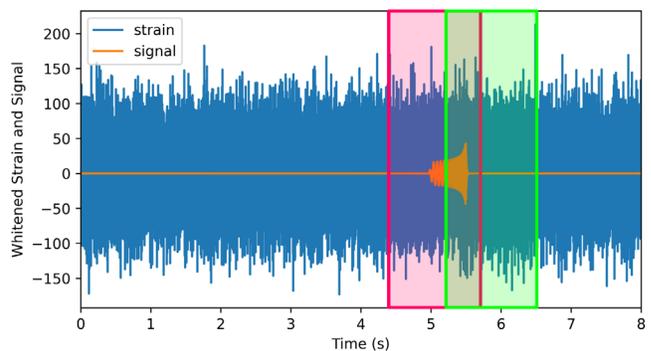
FIG. 3.   Extracting 1-s duration segment from 8-s-long strain. The red and green blocks represent two examples of extraction. The blue curve is a strain that consists of GW waveform plus noise. The orange curve is the theoretical GW waveform in the strain.

$$\text{FAP} = \frac{S_\text{N} \cap S_\text{pGW}}{S_\text{N}}. \qquad (7)$$

When the output of the model p is bigger than a given threshold P_c, the input sample is classified as GW signal $s \in S_\text{pGW}$. In this way S_pGW is affected by the threshold P_c. Along with the changing of P_c, corresponding TAP and FAP can be obtained.

The building process of the base learners is shown in Fig. 4. For each detector (Hanford or Livingston), ten base learners are selected from 100 weak learners, and 100 weak learners are trained by the 100 datasets generated in the previous section. In the training process of the weak learner, 10% of the data in the training set corresponding to the weak learner are randomly selected to verify the weak learner, and we reserve the weak learners with the highest verification accuracy. Note that this part of the validation data does not belong to the validation set used in the next stage, but to the training set corresponding to each weak learner. After getting 100 weak learners, we calculate the AUC of each weak learner based on the validation set. AUC means the area under the ROC curve. AUC has been widely used to reflect the generalization ability of a model [36]. We select the top ten models with the largest AUC in the 100 weak learners as the base learner. The code implementation of the above weak learner training and base learner choice is based on the Keras framework [60]. All of the computations are done on a NVIDIA 1080Ti GPU. Based on such hardware, the typical delay time of our ensemble model is less than 2 miniseconds.

3. Subensemble model construction

As explained before and shown in Fig. 2, our whole ensemble model includes two subensemble models corresponding to Hanford and Livingston detectors. Here we describe how to construct these two subensemble models.

Using the aforementioned method of weak learner training, selecting, and assembling, we get an ensemble model consisting of ten base learners. In this section, the performance of the single Gabbard model (model I) [30], the ensemble model using the arithmetic mean voting strategy (model II), and the ensemble model using the harmonic mean voting strategy (model III) are compared.

Most of the strain data recorded by LIGO do not include GW signal. Even for current LIGO, the recorded data in most of the time are the background noise without signal. Due to the data characteristics with extremely asymmetrical distribution, the main problem in deep learning for GW data analysis is false triggering. In long-duration data, the model may generate numerous false triggers even its false-alarm probability is relatively low. The research of deep learning models with low false-alarm probability is helpful
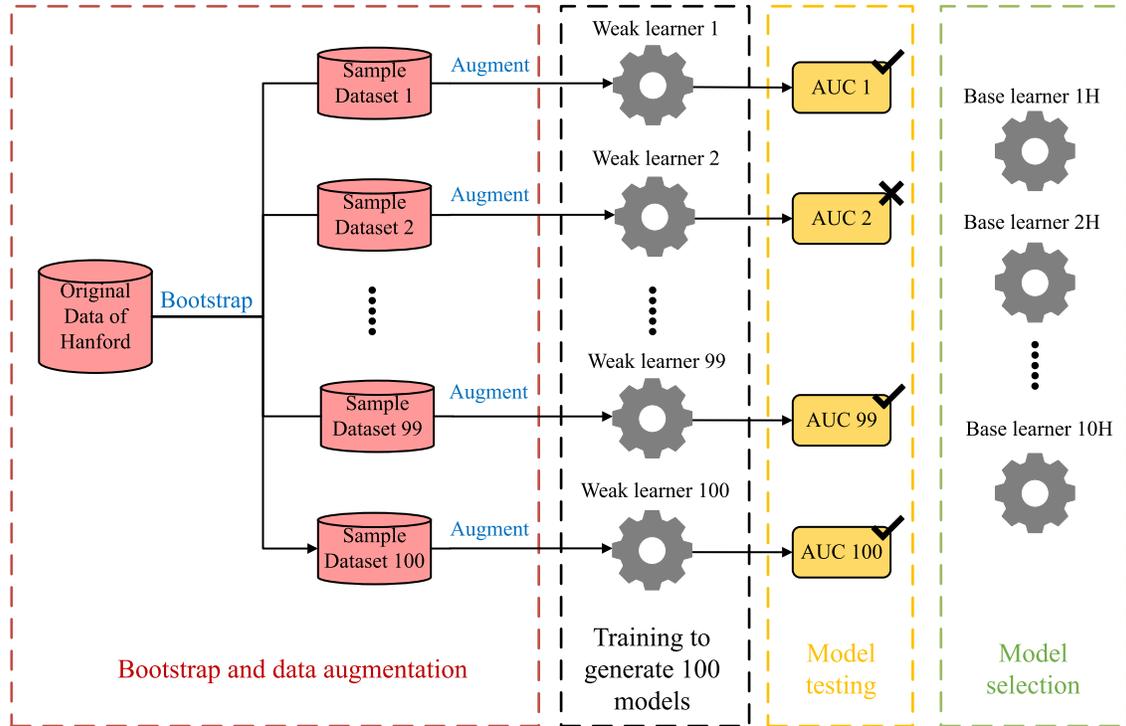
FIG. 4. Schematic diagram of building process of base learners. There are four steps to build the base learners. First step is using bootstrap and data augmentation to build 100 training sets. Second step is training 100 weak learners. Third step is using validation set to calculate the area under curve (AUC) of each weak learner. Final step is selecting top ten weak learners with largest AUC as base learners.
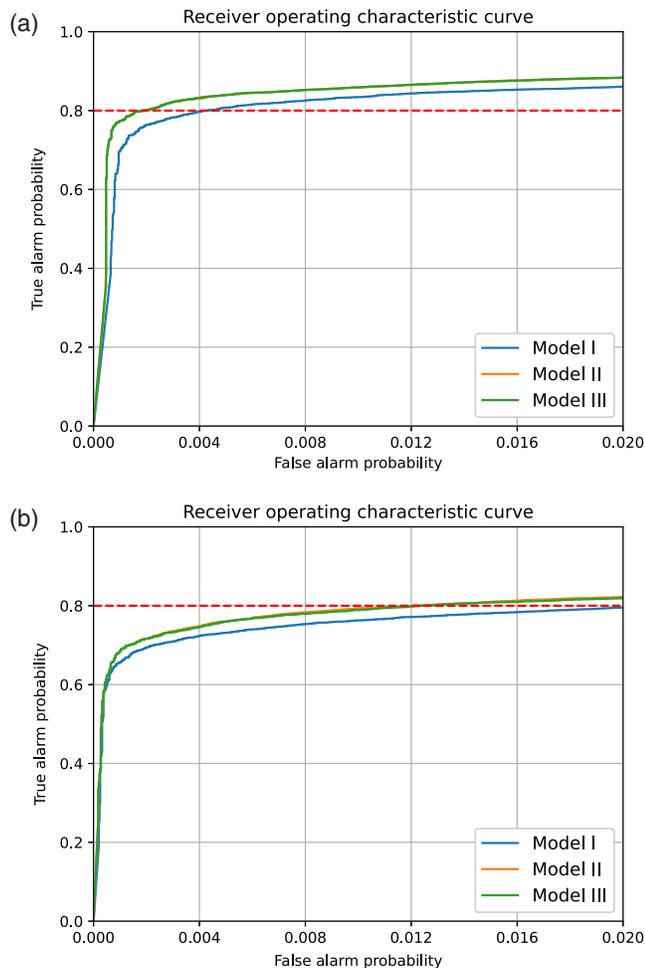
FIG. 5.   ROC curve of subensemble models and single model trained by Hanford (a) and Livingston (b) interferometer data. The TAP of red line is 0.8. The blue, orange, and green lines indicate the ROC curve of model I, II, and III, respectively.

to develop a gravitational wave detection model with a low number of false-alarm triggers. Therefore, we pay much attention to the ROC curves of the low FAP interval.

The ROC curves of model I model II and model III are drawn in the range FAP $\in [0, 0.02]$ which are shown in Fig. 5. The ROC curves of model II and model III are almost the same in this range. For the results on Hanford interferometer [Fig. 5(a)], when the TAP is 0.8, the FAPs of models I, II, and III are 0.0042, 0.00195, and 0.00185, respectively, which indicate that with the same TAP, the FAP of model II and model III are about 54 and 56% lower than that of model I. For Livingston interferometer [Fig. 5(b)], when the TAP is 0.8, the FAPs of models I, II, and III are 0.0025, 0.0012, and 0.0012, respectively, and in this case the FAPs of model II and model III are reduced by about 52% compared to model I.

From the above results, we can see that ensemble models have lower FAP than single models with the same TAP. The use of our ensemble model can greatly reduce the number

of false triggers in real data analysis. Such improvement can make the GW signal recognition more reliable.

4. Moving time step

For real GW detection data analysis, we face a long-duration data. We take a data segment with duration 1 s and plug it into our ensemble model for signal recognition. Afterwards we move forward a time step Δt to take the following data segment to the following works. The time step will also affect the signal recognition.

Specifically, we consider the GW arrival time problem. The gravitational wave arrival time of BBH merger is predicted according to the time range of continuous alarm of GW trigger as

$$t_{\text{arr}} = \frac{1}{2} \times (t_{\text{fe}} + t_{\text{lb}}), \qquad (8)$$

where t_{fe} represents the end time of the detection window of the first alarm and t_{lb} represents the beginning time of the detection window at the last alarm. Since the time position of the GW signal in the training set is symmetrically distributed, we take the midpoint of this interval as the predicted arrival time of the detected gravitational wave. Note that the arrival time (8) does not mean the beginning time of the detected GW signal. Instead, it corresponds to a specific time of the detected GW strain on a given interferometer. Such information is important to be used to determine the location of the GW source.

We use 10,000 samples for each interferometer which contain GW signals in the test set to test the above-mentioned arrival time prediction issue. Based on model III we have tested two moving time steps, $\Delta t = \frac{1}{8}$ s and $\frac{1}{16}$ s. We estimated the prediction error with the difference between the t_{arr} gotten via Eq. (8) and the simulated merger time of the binary black hole coalescence waveform. We count the error and plot the histogram in Fig. 6. With $\Delta t = \frac{1}{8}$ s [Fig. 6(a)] and $\frac{1}{16}$ s [Fig. 6(b)], respectively, model III successfully detects 8711 and 8751 signals for Hanford interferometer and successfully detects 7851 and 7921 signals for Livingston interferometer.

As we can see in Fig. 6(a), among all successfully detected strain in the test on Hanford and Livingston interferometer, 51.1 and 50.6% predictions admit errors falling in $[-\frac{1}{8}, \frac{1}{8}]$, and 98.1 and 98% predictions admit errors falling in $[-\frac{1}{4}, \frac{1}{4}]$. In Fig. 6(b), of all successfully detected strain in the test on Hanford and Livingston interferometer, 71.3 and 70.8% predictions admit errors falling in $[-\frac{1}{8}, \frac{1}{8}]$; 95.7 and 95.8% predictions admit errors falling in $[-\frac{1}{4}, \frac{1}{4}]$.

Above results indicate that the predicted GW arrival time (8) corresponds to the merger time of BBH. We can understand such correspondence in the following way. BBH merger results in the strongest strain for the whole
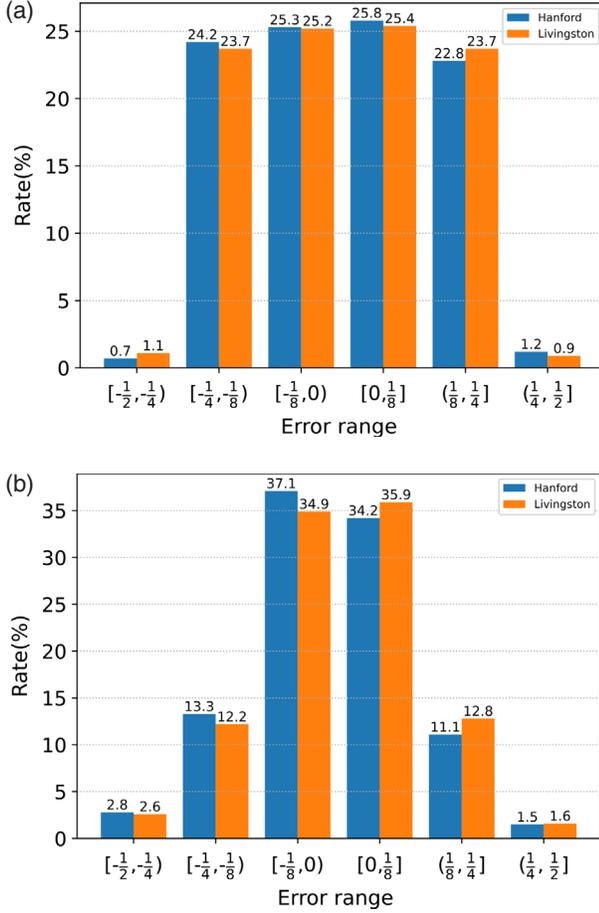
FIG. 6. The distribution histogram of the prediction error of the merger time. Model III is used here. Blue line and orange line are, respectively, for Hanford and Livingston interferometer. (a) and (b) correspond to moving time step $\Delta t = \frac{1}{8}$ s and $\frac{1}{16}$ s, respectively.

coalescence process. Consequently, the merger time will appear at the center of the continuous triggers produced by our ensemble model. Such fact makes the estimated time (8) correspond to the merger time of BBH. In addition we find from the above test results that the accuracy of the predicted merger time of BBH GW is improved when the moving time step Δt changes from $\frac{1}{8}$ s to $\frac{1}{16}$ s. We find this is a general trend that smaller moving time step results in more accurate merger time prediction.

IV. APPLICATION TO O1 AND O2 DATA

Now we apply our ensemble model to the O1 and O2 data of LIGO detectors. As we designed in the last section, the length of data segment window is 1 s. We take the moving time step $\Delta t = \frac{1}{8}$ s.

We have done two investigations. In the first one, we consider all GW events of O1 and O2 reported by LIGO. For each event we take 16-s-long data and let the GW event locate at the center of the corresponding strain. In the second investigation, we consider 68-min-long strain and

also let the GW event locate at the center of the corresponding strain.

A. Threshold setting

GW signal recognition is essentially a binary classification problem. Many researchers use a strategy with a detection threshold of 0.5 [36,37]. The background noise related to Hanford and Livingston interferometers has different statistical property. Consequently, the models trained by different detector noise admit different character. With respect to the same threshold, two ensemble models working for Hanford and Livingston interferometers, respectively, have different FAP and TAP. Accordingly, we chose different thresholds for the two ensemble models.

We decide the threshold of the two ensemble models through fixing the FAP to 0.004. Through the ROC curve obtained above, we calculate the thresholds corresponding to model I, model II, and model III, respectively. The thresholds are shown in the Table II. When the output of a subensemble model exceeds its corresponding threshold, an alarm will be given.

B. Performance of subensemble models

We firstly apply the subensemble model to the single-interferometer data. When the number of continuous alarms exceeds five, we let the model generate a GW trigger.

Regarding the 16-s-strain tests, we plot the analysis results in Fig. 7. H denotes the strain detected by Hanford and L denotes the strain detected by Livingston. The comparison among models I, II, and III is shown there. Except H-GW170818, all BBH GW events in O1 and O2 are successfully recognized by ensemble model II and model III. As we mentioned before, although the ensemble models are trained based on only O1 data, they work very well for O2 data. In contrast, the single model I misses H-GW151226, L-GW151226, and L-GW170608 signals. Since the training data do not contain binary neutron star (BNS) coalescence, the BNS gravitational wave event GW170817 is lost. As we can see from the recognition results of H-GW170817, all three types of models produced outputs greater than 0.5 at 1 s, but the duration of the outputs did not cover the time of the event and the outputs did not exceed the corresponding thresholds. Similarly, for L-GW170817, although the three types of models have successfully generated GW triggers, the duration of the trigger is more matched with the occurrence time of the

TABLE II. The thresholds of the three models on Hanford and Livingston interferometers.

Detector	Hanford	Livingston
Model I	0.9761	0.9421
Model II	0.8877	0.8793
Model III	0.8363	0.8454
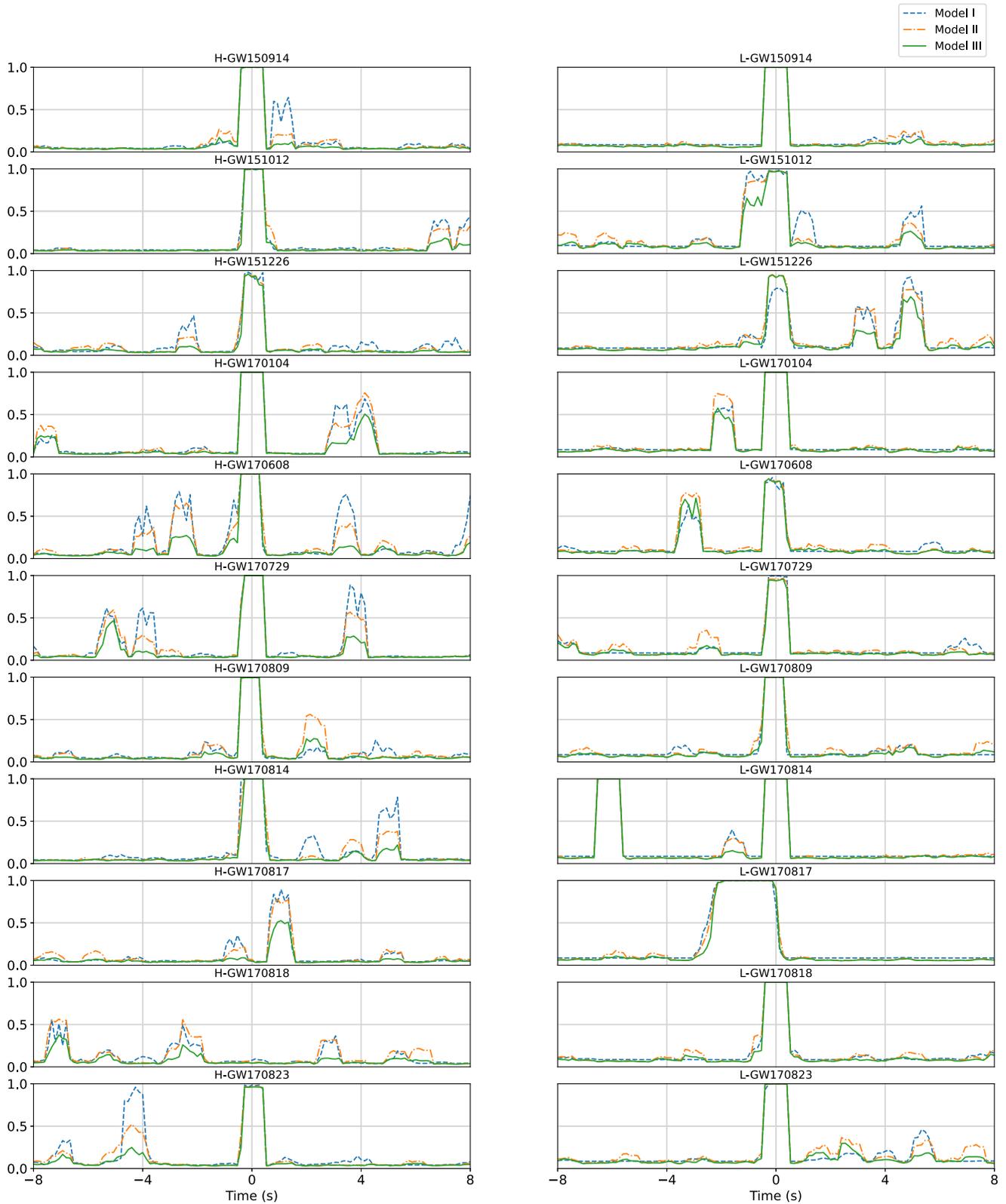
FIG. 7.   The GW signal recognition results for the events of O1 and O2. The blue, orange, and green lines indicate the response of models I, II, and III, respectively. H and L represent the tests done for Hanford and Livingston interferometers, respectively. All GW events happened at 0 s.

(a)
GW170814 detected by Livingston detector

(b)
GW170817 detected by Livingston detector
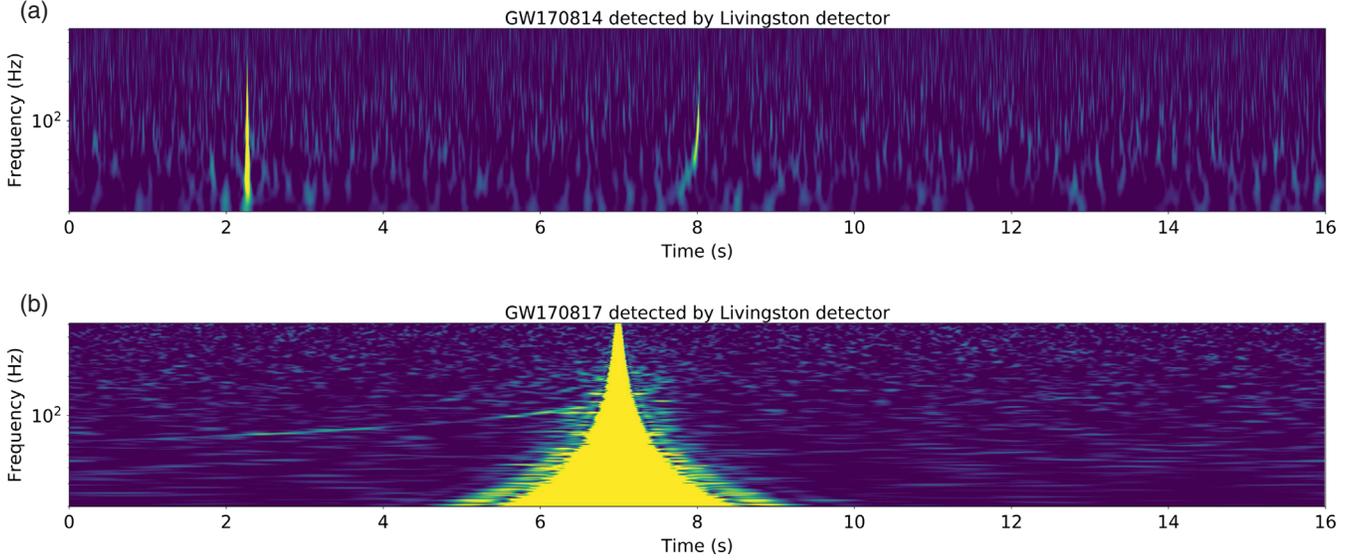
FIG. 8. The spectrogram of 16-s strain near L-GW170814 and L-GW170817. Each strain contain a glitch which results in the false trigger of the ensemble models shown in Fig. 7. The GW events happened at 8s. Two types of glitch happened near 2 s and 7 s in (a) and (b), respectively.

glitch shown in Fig. 8(b). We think this trigger may be a false trigger.

Interestingly, all models result in a false trigger about 6 s before the event for L-GW170814. We check the constant-Q Transformation [61] of L-GW170814 in Fig. 8(a). From this figure, we can see a glitch corresponding to the false trigger near L-GW170814.

Similarly, all models result in a false trigger with an alarm time more than 2 s at approximately 4 s before the GW event L-GW170817. We again check the constant-Q Transformation of L-GW170817 in Fig. 8(b). We can see the famous glitch of GW170817 which results in the false trigger.

Regarding H-GW170818, none of the models successfully recognize this signal. We remind that the matching SNR of GW170818 in Hanford is only 4.1. GW170818 was only detected by GstLAL [62,63]. PyCBC [64] missed this signal because the SNRs in Hanford and Virgo are 4.1 and 4.2, respectively, which was lower than the threshold 5.5 [65] used by PyCBC. Similarly, the training set used in the current work admitted the lowest SNR 7. Consequently, our model also missed this signal.

We have seen above that the subensemble models perform better than the single model. In the next subsection we will see that our whole ensemble model can even improve much more.

C. Performance of the whole ensemble model

The test results shown in the last subsection show that the subensemble models corresponding to a single interferometer behaves better than a single model. Now we combine the two subensemble models for Hanford and Livingston interferometers to obtain more reliable real-time GW signal

recognition results. In this subsection, we even increase the difficulty of the problem. We consider 68-min-long data, which are much longer than the 16 s considered in the last subsection. The schematic diagram of our ensemble model's work style is shown in Fig. 9. If the two subensemble models generate more than five alerts continuously, a GW signal trigger will be given. All the tests done in this subsection are based on model III.

Regarding the false-alarm problem, the two individual subensemble models produce averagely 12.85 and 11.07 triggers per hour, respectively, for Hanford and Livingston interferometers. In contrast, our ensemble model does not give triggers at all except the ten BBH events. Regarding the missing trigger problem, our ensemble model fails to find out GW170818. As we analyzed in the last subsection, GW170818 is only detected by the Livingston interferometer. Consequently, our ensemble model misses this signal.

Besides the 68-min data segment near the ten BBH events, we tested the whole LIGO data in August 2017. Note that we only analyzed the time when both Hanford and Livingston interferometers were normally working. After combining the two submodels, our whole ensemble model does not give any more triggers except the GW events reported by LIGO.

The results of the FAP do not directly correspond to the false-alarm rates (FAR) on continuous detection via moving the time window. In the continuous analysis of background noise, the FAR reflects the average time for the model generating a false trigger. The FAR is given by [66]

$$\text{FAR} = \frac{N_f}{t_0}, \qquad (9)$$

Strains from Hanford interferometer — **Ensemble model 1** — Working on Hanford detector

Strains from Livingston interferometer — **Ensemble model 2** — Working on Livingston detector

Two models response to gravitational waves — Generated by total detection model — GW trigger
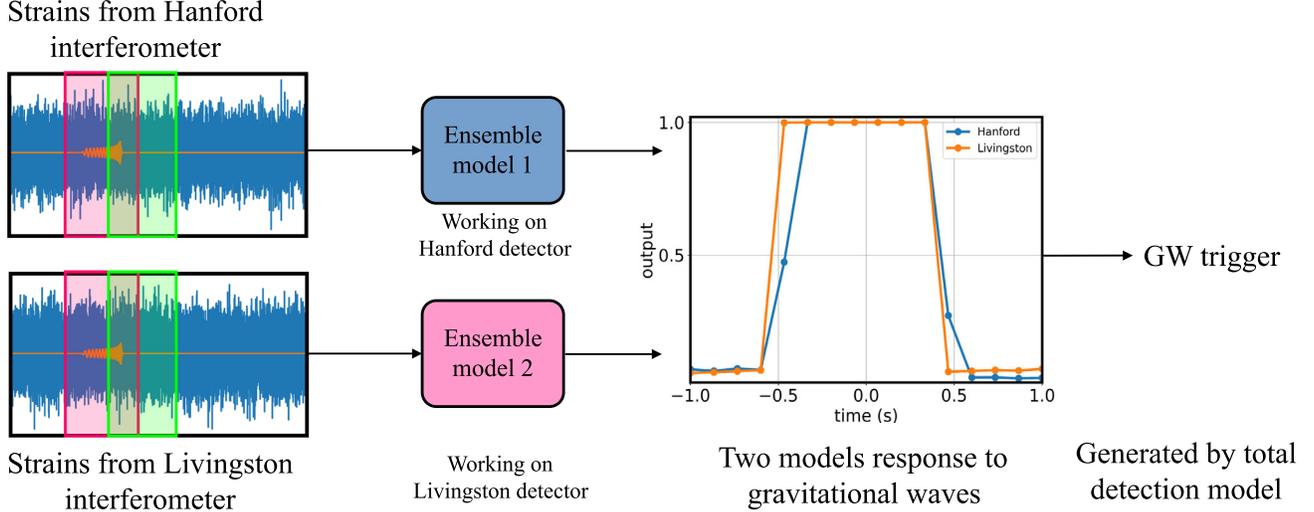
FIG. 9. The schematic diagram of the ensemble model's work style. When the two subensemble models generate more than five alerts continuously, a GW signal trigger is given by the ensemble model.

where N_f represents the number of false triggers generated in detection process and t_0 represents the detection duration. However, there was no false trigger in the one-month detection by using the whole detection model; we cannot estimate the FAR of the whole ensemble model directly. Alternatively, we can estimate the FAR from the detection results of two subensemble models. From Eq. (9), we can get

$$\text{FAR} = \frac{N_f/N}{t_0/N} = \frac{p_{ft}}{d_s}, \tag{10}$$

where N represents the number of detections in the duration t_0, $p_f = \frac{N_f}{N}$ corresponds to the probability of generating a false trigger and $d_s = \frac{t_0}{N}$ corresponds to the detection time step. Note that p_f is not FAP. FAP only considers the output of a single detection rather than GW trigger (more than five continuous alerts). Assuming that Hanford and Livingston interferometers have independent distributions of background noise data, we can conclude that

$$p_f = p_{fH} \times p_{fL}, \tag{11}$$

where p_{fH} and p_{fL} represent the probability of generating a false trigger by the subensemble model working on Hanford and Livingston interferometers, respectively. Based the August 2017 detection result of two subensemble models, we find that $p_{fH} \approx 4.16 \times 10^{-4}$ and $p_{fL} \approx 3.09 \times 10^{-4}$. Through Eq. (11), $p_f \approx 1.29 \times 10^{-7}$. Consequently, the whole ensemble model's FAR is about 1 per 11.24 d. The co-working time of Hanford and Livingston interferometers in August 2017 is about 17 d, which means the expected false-alarm event is about 1. That is consistent with the fact that there is no false trigger happening at all.

We also estimate the arrival time of the three BBH merger events by Eq. (8) and list the result in Table III. We

find that the prediction error of the three GW events is less than 0.1 s. This means our ensemble model can roughly recover the data analysis results based on complicated Bayesian method. Based on these good performances, we conclude that our ensemble model can be well applied to real-time GW signal recognition.

V. SUMMARY

In this work, we designed an ensemble model for gravitational wave signal recognition. The whole ensemble model consists of two individual subensemble models. The model is trained by O1 data of Hanford and Livingston interferometers, respectively. We applied this ensemble model to the gravitational wave events of O1 and O2 of LIGO. Our ensemble model can successfully figure out nine BBH GW events of the ten reported ones by LIGO. The missed GW170818 admits subthreshold behavior for Hanford detector. This means our ensemble model admits very high true-alarm rate.

We have also applied our ensemble model to the whole-month LIGO data in August 2017. Except the BBH events reported by LIGO, our ensemble model does not give extra GW trigger. This means our ensemble model admits extremely low false-alarm rates. These test results indicate that the machine learning with ensemble algorithm [44] can greatly reduce the number of false triggers in the GW signal recognition process.

TABLE III. The merge time prediction.

Event	Predict merge time	Merge time [67]
GW170809	1186302519.75	1186302519.8
GW170814	1186741861.5625	1186741861.5
GW170823	1187529256.5625	1187529256.5

The above-mentioned high true-alarm rate and extremely low false-alarm rate prove that the optimized ensemble learning algorithm can be applied to real-time gravitational wave signal recognition. Compared with feature fusion strategy, cross validation used in the ensemble model does not require additional training and consequently less computation cost. And, cross validation is more explicable than feature fusion.

As caveats, there are also drawbacks to cross validation. Some GW signal may be confidently detected by just one detector while the other detector admits low SNR. Such kind of signal will be missed like GW170818.

The test code and the source data are available at https://github.com/WeiWang31/EnsembleModel.

[1] B. P. Abbott, R. Abbott, T. Abbott, M. Abernathy, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, R. Adhikari *et al.*, Phys. Rev. Lett. **116,** 061102 (2016).

[2] B. P. Abbott, R. Abbott, T. Abbott, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, R. Adhikari, V. Adya *et al.*, Phys. Rev. Lett. **119,** 161101 (2017).

[3] B. P. Abbott, R. Abbott, T. Abbott, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, R. Adhikari, V. Adya *et al.*, Astrophys. J. Lett. **848,** L13 (2017).

[4] A. Goldstein, P. Veres, E. Burns, M. Briggs, R. Hamburg, D. Kocevski, C. Wilson-Hodge, R. Preece, S. Poolakkil, O. Roberts *et al.*, Astrophys. J. Lett. **848,** L14 (2017).

[5] V. Savchenko, C. Ferrigno, E. Kuulkers, A. Bazzano, E. Bozzo, S. Brandt, J. Chenevez, T.-L. Courvoisier, R. Diehl, A. Domingo *et al.*, Astrophys. J. Lett. **848,** L15 (2017).

[6] L. S. Collaboration, V. Collaboration, G. Fermi, INTERGRAL, and Y. C. Hello, Astrophys. J. **848,** L12 (2017).

[7] B. Allen, Phys. Rev. D **71,** 062001 (2005).

[8] C. Robinson, B. S. Sathyaprakash, and A. S. Sengupta, Phys. Rev. D **78,** 062002 (2008).

[9] E. A. Huerta, G. Allen, I. Andreoni, J. M. Antelis, E. Bachelet, G. B. Berriman, F. B. Bianco, R. Biswas, M. C. Kind, K. Chard *et al.*, Nat. Rev. Phys. **1,** 600 (2019).

[10] K. Fukushima and S. Miyake, in *Competition and Cooperation in Neural Nets* (Springer, New York, 1982), pp. 267–285.

[11] T. D. Sanger, Neural Netw. **2,** 459 (1989).

[12] K. Fukushima, S. Miyake, and T. Ito, IEEE Trans. Syst. Man Cybernet. **13,** 826 (1983).

[13] R. Hecht-Nielsen, in *Neural Networks for Perception* (Elsevier, New York, 1992), pp. 65–93.

[14] Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel, *Advances in Neural Information Processing Systems*, edited by D. Touretzky (Morgan-Kaufmann, Denver, Colorado, 1989), Vol. 2, pp. 396–404.

[15] S. Hochreiter, Int. Uncertain. Fuzziness Knowl.-Based Syst. **06,** 107 (1998).

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, *Advances in Neural Information Processing Systems*, edited by F. Pereira, C. J. Burges, L. Bottou, and K. Q. Weinberger (Curran Associates, Inc., Lake Tahoe, Nevada, 2012), Vol. 25, pp. 1097–1105.

[17] K. He, X. Zhang, S. Ren, and J. Sun, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, Las Vegas, Nevada, 2016), pp. 770–778.

[18] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, Las Vegas, Nevada, 2016), pp. 2818–2826.

[19] K. Simonyan and A. Zisserman, arXiv:1409.1556.

[20] J. Long, E. Shelhamer, and T. Darrell, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, Boston, Massachusetts, 2015), pp. 3431–3440.

[21] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, IEEE Trans. Pattern Anal. Mach. Intell. **40,** 834 (2017).

[22] K. He, G. Gkioxari, P. Dollár, and R. Girshick, in *Proceedings of the IEEE International Conference on Computer Vision* (IEEE, Venice, 2017), pp. 2961–2969.

[23] C. Szegedy, A. Toshev, and D. Erhan, *Advances in Neural Information Processing Systems*, edited by C. J. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger (Curran Associates, Inc., Lake Tahoe, Nevada, 2013).

[24] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, Las Vegas, Nevada, 2016), pp. 779–788.

[25] S. Ren, K. He, R. Girshick, and J. Sun, IEEE Trans. Pattern Anal. Mach. Intell. **39,** 1137 (2017).

[26] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, IEEE Signal Process. Mag. **29,** 82 (2012).

[27] A. Graves, A.-r. Mohamed, and G. Hinton, in *Proceedings of the 2013 IEEE International Conference on Acoustics,*

Speech and Signal Processing (IEEE, New York, 2013), pp. 6645–6649.

[28] A. Graves and N. Jaitly, in *International Conference on Machine Learning* (PMLR, Beijing, 2014), pp. 1764–1772.

[29] D. George and E. A. Huerta, Phys. Lett. B **778,** 64 (2018).

[30] H. Gabbard, M. Williams, F. Hayes, and C. Messenger, Phys. Rev. Lett. **120,** 141103 (2018).

[31] M. L. Chan, I. S. Heng, and C. Messenger, Phys. Rev. D **102,** 043022 (2020).

[32] G. Baltus, J. Janquart, M. Lopez, A. Reza, S. Caudill, and J.-R. Cudell, Phys. Rev. D **103,** 102003 (2021).

[33] B. Beheshtipour and M. A. Papa, Phys. Rev. D **103,** 064027 (2021).

[34] R. Ormiston, T. Nguyen, M. Coughlin, R. X. Adhikari, and E. Katsavounidis, Phys. Rev. Research **2,** 033066 (2020).

[35] C. Chatterjee, L. Wen, K. Vinsen, M. Kovalam, and A. Datta, Phys. Rev. D **100,** 103025 (2019).

[36] H. Wang, S. Wu, Z. Cao, X. Liu, and J.-Y. Zhu, Phys. Rev. D **101,** 104003 (2020).

[37] H. Xia, L. Shao, J. Zhao, and Z. Cao, Phys. Rev. D **103,** 024040 (2021).

[38] H. Wang, Z. Cao, Y. Zhou, Z.-K. Guo, and Z. Ren, Big Data Min. Anal. **5,** 53 (2022), https://ieeexplore.ieee.org/abstract/document/9663260.

[39] J. Yan, M. Avagyan, R. E. Colgan, D. Veske, I. Bartos, J. Wright, Z. Márka, and S. Márka, Phys. Rev. D **105,** 043006 (2022).

[40] A. Menéndez-Vázquez, M. Kolstein, M. Martinez, and L. M. Mir, Phys. Rev. D **103,** 062004 (2021).

[41] B. V. Dasarathy and B. V. Sheela, Proc. IEEE **67,** 708 (1979).

[42] L. K. Hansen and P. Salamon, IEEE Trans. Pattern Anal. Mach. Intell. **12,** 993 (1990).

[43] W. Wei, A. Khan, E. Huerta, X. Huang, and M. Tian, Phys. Lett. B **812,** 136029 (2021).

[44] E. Huerta, A. Khan, X. Huang, M. Tian, M. Levental, R. Chard, W. Wei, M. Heflin, D. S. Katz, V. Kindratenko *et al.*, Nat. Astron. **5,** 1062 (2021).

[45] L. Breiman, Mach. Learn. **24,** 123 (1996).

[46] T. D. Gebhard, N. Kilbertus, I. Harry, and B. Schölkopf, Phys. Rev. D **100,** 063015 (2019).

[47] Y. Pan, A. Buonanno, A. Taracchini, L. E. Kidder, A. H. Mroué, H. P. Pfeiffer, M. A. Scheel, and B. Szilágyi, Phys. Rev. D **89,** 084006 (2014).

[48] A. Bohé, L. Shao, A. Taracchini, A. Buonanno, S. Babak, I. W. Harry, I. Hinder, S. Ossokine, M. Pürrer, V. Raymond *et al.*, Phys. Rev. D **95,** 044028 (2017).

[49] Z. Cao and W.-B. Han, Phys. Rev. D **96,** 044028 (2017).

[50] X. Liu, Z. Cao, and L. Shao, Phys. Rev. D **101,** 044049 (2020).

[51] C. Cutler and E. E. Flanagan, Phys. Rev. D **49,** 2658 (1994).

[52] X.-R. Li, W.-L. Yu, X.-L. Fan, and G. J. Babu, Front. Phys. **15,** 1 (2020).

[53] B.-J. Lin, X.-R. Li, and W.-L. Yu, Front. Phys. **15,** 1 (2020).

[54] P. G. Krastev, K. Gill, V. A. Villar, and E. Berger, Phys. Lett. B **815,** 136161 (2021).

[55] X. Fan, J. Li, X. Li, Y. Zhong, and J. Cao, Sci. China **62,** 1 (2019).

[56] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, J. Mach. Learn. Res. **15,** 1929 (2014).

[57] T. Dozat, *Incorporating Nesterov Momentum into Adam* (ICLR, San Juan, Puerto Rico, 2016), https://openreview.net/pdf?id=OM0jvwB8jIp57ZJjtNEZ.

[58] B. Efron, in *Breakthroughs in Statistics* (Springer, New York, 1992), pp. 569–593.

[59] T. Fawcett, Pattern Recogn. Lett. **27,** 861 (2006).

[60] A. Géron, *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems* (O'Reilly Media, Sebastopol, California, 2019).

[61] J. C. Brown, J. Acoust. Soc. Am. **89,** 425 (1991).

[62] S. Sachdev, S. Caudill, H. Fong, R. K. Lo, C. Messick, D. Mukherjee, R. Magee, L. Tsukada, K. Blackburn, P. Brady *et al.*, arXiv:1901.08580.

[63] C. Messick, K. Blackburn, P. Brady, P. Brockill, K. Cannon, R. Cariou, S. Caudill, S. J. Chamberlin, J. D. Creighton, R. Everett *et al.*, Phys. Rev. D **95,** 042001 (2017).

[64] S. A. Usman, A. H. Nitz, I. W. Harry, C. M. Biwer, D. A. Brown, M. Cabero, C. D. Capano, T. Dal Canton, T. Dent, S. Fairhurst *et al.*, Classical Quantum Gravity **33,** 215004 (2016).

[65] B. Abbott, R. Abbott, T. Abbott, S. Abraham, F. Acernese, K. Ackley, C. Adams, R. Adhikari, V. Adya, C. Affeldt *et al.*, Phys. Rev. X **9,** 031040 (2019).

[66] M. B. Schäfer, O. c. v. Zelenka, A. H. Nitz, F. Ohme, and B. Brügmann, Phys. Rev. D **105,** 043002 (2022).

[67] R. Abbott *et al.*, SoftwareX **13,** 100658 (2021).