Oral presentation draft

Begin：

Good morning, everyone. I'm glad to make this presentation. My name is Shougui Cai, from Zhejiang University, China, and the co-author is Professor Wen Xu. The title of our paper is "Matched-field source localization using sparsely-coded neural network and data-model mixed training".

Source localization is a basic problem in underwater acoustics. Many solving approaches have been developed, and the matched-field processing (MFP) is one of the mostly-studied. However, MFP is sensitive to the mismatch problem and performs well only when the knowledge of ocean environment is accurate.

Machine learning learns directly from the observation and can be designed to learn a generic model suitable for different scenarios. In our paper, source localization is viewed as a machine learning problem and a matched-field source localization model is learned by training a sparsely-coded feed-forward neural network (SCFNN) with mixed environment models and data.

Contents：

My presentation will go on in this order. First, i'll introduce how to establish an SCFNN source localization model and how to learn the parameter from data. Then, i'll train and test a localization prediction model on the experimental data, to confirm that the SCFNN works well on source localization. And in part III, the influence of SSP mismatch on the SCFNN model is investigated by simulations, together with performance comparison with two conventional MFP methods, in part IV, data-model mixed training strategy is used to improve the model robustness, the end is a summary & future work.

SCFNN model：

Our work mainly refers to Niu's paper published on JASA (Journal of the Acoustical Society of America), you can find it by search the paper name given below. Same as Niu (UCSD) did in his work (JASA 2017), we also assumed that there is a deterministic [dɪˌtɜːmɪ'nɪstɪk] relationship between source range and sample-covariance matrix (SCM) and approximated this relationship by the FNN.

The feed-forward neural network (FNN), also known as multi-layer perceptron, is constructed using a feedforward directed acyclic [eɪ'saɪklɪk] architecture. The outputs are formed through a series of functional transformations of the weighted inputs.

Here, a three layer model (input layer L1, hidden layer L2 and output layer L3) is used to construct the FNN. The input layer L1 is comprised of D input variables x. The output y is discrete [dɪ'skrit] corresponding [.kɔrə'spɑndɪŋ] to the estimated source range.

Network structure:

The FNN used in our paper is a simple three layer model (input layer L1, hidden layer L2 and output layer L3).

The hidden layer is a linear combination with input layer and then transformed using an activation function f(·). ReLu (Rectified ['rektə.faɪ] Linear Units) function was chosen as the intermediate [.ɪntər'midiət] activation function for this neural network, function curve can been seen in Fig(b).

Similarly, Neurons between the hidden layer and the output layer are mapped by a linear function, and is normalized by *softmax* function, which is a common choice for multi-class classification task.

Input data preprocessing：
The sound pressure is modeled as source term S(f) multiply Green’s function g(f, r), then add with noise term n.

To reduce the effect of the source amplitude |S(f)|, this complex pressure is normalized, and the sample covariance matrices (SCMs) are averaged over Ns snapshots to form the conjugate ['kɑndʒə.geɪt] symmetric [sɪ'metrɪk] matrix.

Only the real and imaginary parts of the complex valued entries of diagonal [daɪ'æɡənəl] and upper triangular matrix in C(f) are used as input.

Preprocessing the data according to these equations ensures that the Green’s function is used for localization.

Source range mapping：
In this probelm, a set of source ranges is discretized [dɪ'skrit] into K bins, r1, ..., rK, of equal width [wɪtθ] Δr.

Here, rk, r = 1,…,K are the source range classes.

Training criterion：
The weight matrix W ,V and bias vector b1, b2 are the parameters to be learned. In our case, the parametric [ˌpærə'metrɪk] model defines a distribution of target location, and we can simply use the principle of maximum likelihood to determine the parameters in this model.

The first term in *E(w)* is the cross-entropy, equivalently [ɪk'wɪvələntlɪ] the negative log-likelihood, between the true/desired distribution and the model predict distribution.

The second and third term are the sparsity constraints [kən'streɪnt] on neural networks.

In this paper, we use L1-norm to promote sparse neurons activations, and constrain the L2-norm of each row of the weight matrix V to prevent any one hidden unit from having very large weights. Thus, the neural network is sparsely coded, we named it as SCFNN.

A sparsely-coded neural network encodes each input data as a sparse code firstly, and then accomplish the specific task with further processing.

By minimizing this criteria [kraɪ'tɪriən], we can learn the model weights from training data and finally get a source localization prediction model.

As the maximum likelihood criterion [kraɪ'tɪrion] is consistent, the model is capable of representing the training data distribution.

Definition of model accuracy：
The model accuracy is defined as the percentage of accurately predicted samples.
In this equation, tn is the label of data xn (x subscript ['sʌbskrɪpt] n). The upper case N is the number of test samples.

Simulation environment：
Here are two picture introduced SWell96-Ex test, conducted in a shallow water waveguide environment with depth of 216 m.

During the experiment, two moving sound sources are deployed in field, including a deep source (J-15) and a shallow source (J-13).

The source ship (R/V Sproul) started its track south of all of the arrays and proceeded northward at a speed of 5 knots (2.5 m/s), as Fig. (b) shows.

The number of vertical array elements is 21, other specific deployment parameters are shown in Fig. (a).

In all of our discussions, the shallow sound source is used, which was towed about 9 m in depth and emitted with 9 frequencies between 109 Hz and 385 Hz.

Performance comparison：
In this part, the proposed model is tested on experimental data, and compared with two methods denoted as Bartlett and MCE, Bartlett use the measured pressure to match with a replica field computed by a numerical propagation model, while, MCE matches the covariance. Note that, there are two kinds of replica-field used in the Bartlett processor, one is simulated by Kraken (noted as Bartlett 2), the other is the measurement data (noted as Bartlett 1), same as the training data used in SCFNN.

As we can see, whether using single frequency or multi-frequencies, the accuracy of SCFNN is always better than the Bartlett, and not worse than direct data match (noted as MCE),.

This is more obvious when it comes to the comparison of absolute mean error. It can be said that, the learned SCFNN works well on source localization and performs better than the Bartlett processor.

SSP mismatch on performance：
Here two different degrees of error (a large one and a light one) in the knowledge of the sound-

speed profile are chosen to investigate how the SSP mismatch influence the model performance.

The optimized one is the best SSP model for real environment of SWellEx-96 experiment, while, i906 has significant change in shape from the optimized, while the change in the i905 is slight. The i906* is slightly changed from i906, for the sake of testing.

The performance curves for SCFNN, Bartlett and MCE vs SNR are plotted by 1000 time Monte Carlo simulations. In the Fig., the legend 'FNN, i905' means that, the corresponding method is FNN and the test data is from i905 environment, rests are similar. The snapshot number here is 10.

Results show that when the change in SSP is relatively slight, SCFNN positions best, followed by MCE and Bartlett worst.

When the change is relatively large (with shape varying), the accuracy order is unchanged, but the absolute mean error of SCFNN becomes larger than MCE. This is maybe caused by the noisy training data.

Compared with the performance on ssp-i905, we can see that, when the environment SSP has a big change in shape, the SCFNN performs poorly, and the accuracy drops about 40%.

We can say that, SCFNN is also sensitive to SSP mismatch, but still performs better than Bartlett and the performance of SCFNN is close to the MCE method.

Data-model mixed training：

As neural networks are strong enough to learn regular pattern ['pætərn] over a set of changing scenarios [səˈnɛrioʊ], when training the network, we can use the examples gathered from different mismatch scenarios to make the network be robust to mismatch.

In this section, by combining the data collected from ssp-i906 and ssp-optimized as training set, the robustness of the classifier increases significantly;

By data-model mixed trained, the re-trained classifier predicts accurately on ssp-i906*, just as well as on ssp-i905. Although the accuracy for i905 has a little glissade compared with data training only case, the performance for i906 is improved.

In Fig., the legend 'i905, combined' means the model is trained by mixed data, and then tested on ssp-i905. The rest legends are similar.

We can say that, by using mixed data-model training, the SCFNN classifier can work well on two entirely different SSPs.

Summary & Future work：

In conclusion, Machine learning has potential advantages in unstable underwater acoustic environment.

Combined with data-model mixed training, the SCFNN model can help reduce the mismatch problem in matched-field source localization.

SCFNN learns a sparse and low rank model, needs fewer basis functions to span the data space and is beneficial to capture characteristic of data distributions, which will make the model more descriptive, in our example, averagely, each data sample can be represented by only 16 features.

For now, the discussions are preliminary [prɪ'lɪmə.neri]. It deserves more efforts.

Our future work may include the following orientations [.ɔriən'teɪʃ(ə)n].

Firstly, as the localization error at low SNR is still huge, we will integrate ['ɪntə.greɪt] adversarial [.ædvɜr'seriəl] learning into our model, and make the characteristic [.kerəktə'rɪstɪk] parameters [pə'ræmɪtər] can be enhanced, when the noise is high.

Secondly, we will try to do some mathematical analysis on the learned model and try to explain how does the model robustness been improved by mixed training.

Thank you for your listening.