2023 2nd International Conference on Robotics, Artificial Intelligence and Intelligent Control (RAIIC)

Novel Regularization Method Exploiting Mutual Information for Deep Neural Networks

Zichen Song*
Lanzhou University
Lanzhou University School of Information Science & Engineering
Lanzhou, Gansu
* Corresponding author: 3528630668@qq.com

Ziyang Chen
Dalian University of Technology
International School of Information Science & Engineering
Dalian, Liaoning
Chenzy20030@163.com

Shengjie Xia
Lanzhou University
Lanzhou University School of Information Science & Engineering
Lanzhou, Gansu
Xiasji21@lzu.edu.cn

***Abstract*—Dropout is a powerful way for preventing model overfitting. However, it is inefficient due to it randomly ignoring some neurons. Although there are many ways on Dropout, they are still either inefficient on improving generalization ability or not effective enough. In this paper, we propose Mutual Information Dropout, which is an efficient Dropout based on dropping neurons with low mutual information. In Mutual Information Dropout, instead of randomly ignoring some neurons, we first evaluated the mutual information of neurons to dropout with mutual information below a certain threshold. In this way, Mutual Information Dropout can achieve effective improving generalization ability with evaluate neurons. Extensive experiments on Three datasets show that Mutual Information Dropout is much more efficient than many existing Dropout and can meanwhile achieve comparable or even better generalization ability.**

Keywords-Dropout; Mutual Information; Generalization Ability;

I. INTRODUCTION

Dropout and their variants have achieved great success in many fields. For example, preventing overfitting in neural networks, improving model accuracy in image classification, balancing exploration and exploitation in reinforcement learning and so on. The core of a dropout is randomly ignoring some neurons, which allows the dropout to improving generalization ability. However, since dropout discards some useful information during the training process. Thus, it is difficult for standard dropout to efficiently improve generalization ability.

There are many ways to improve the generalization ability of neural networks, such as Drop-Connect, which sets each weight of the network to 0 with a certain probability during each training iteration. However, this way incurs significant computational overhead, and it can be difficult to select an appropriate dropout rate. Another way, Inverted Dropout, sets each neuron to zero with probability p and then divides its output by p during training. However, it can also be difficult to select an appropriate dropout rate, and these ways may not be efficient if an inappropriate dropout rate is chosen.

In this paper we propose Mutual Information Dropout, which is an efficient Dropout variant based on dropping neurons with low mutual information that can achieve effective generalization ability. In Mutual Information Dropout, we first use mutual information to evaluate the usefulness of neurons. Next, we set an appropriate mutual information threshold. Finally, we perform neuron dropout on those neurons that fall below a certain threshold. We conduct extensive experiments on three benchmark datasets in various tasks. The results demonstrate that Mutual Information Dropout is much more efficient than many Dropout ways and can achieve improving the generalization ability.

The contributions of this paper are summarized as follows:

- We propose a Dropout based on dropping neurons with low mutual information. And we discussed the feasibility of applying mutual information to Dropout.
- We propose a method to explore the performance of neural networks and demonstrate this idea in experiments.
- Extensive experiments on three datasets show that Mutual Information Dropout is much more efficient than many Dropout ways and can achieve competitive performance.

II. MUTUAL INFORMATION DROPOUT

In this section, we introduce our Mutual Information Dropout approach based on dropping neurons with low mutual information. The architecture of Mutual Information Dropout is shown in Fig. 1. It first uses mutual information to evaluate the usefulness of neurons, next set an appropriate mutual information threshold, and finally perform the neuron dropout on those neurons that fall below a certain threshold. In this way, the test performance of the recompiled model was significantly better than that of the previously trained model and have effective generalization ability. Next, we introduce the details of Mutual Information Dropout in the following section.

2023 2nd International Conference on Robotics, Artificial Intelligence and Intelligent Control (RAIIC) | 979-8-3503-2800-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/RAIIC59453.2023.10280939

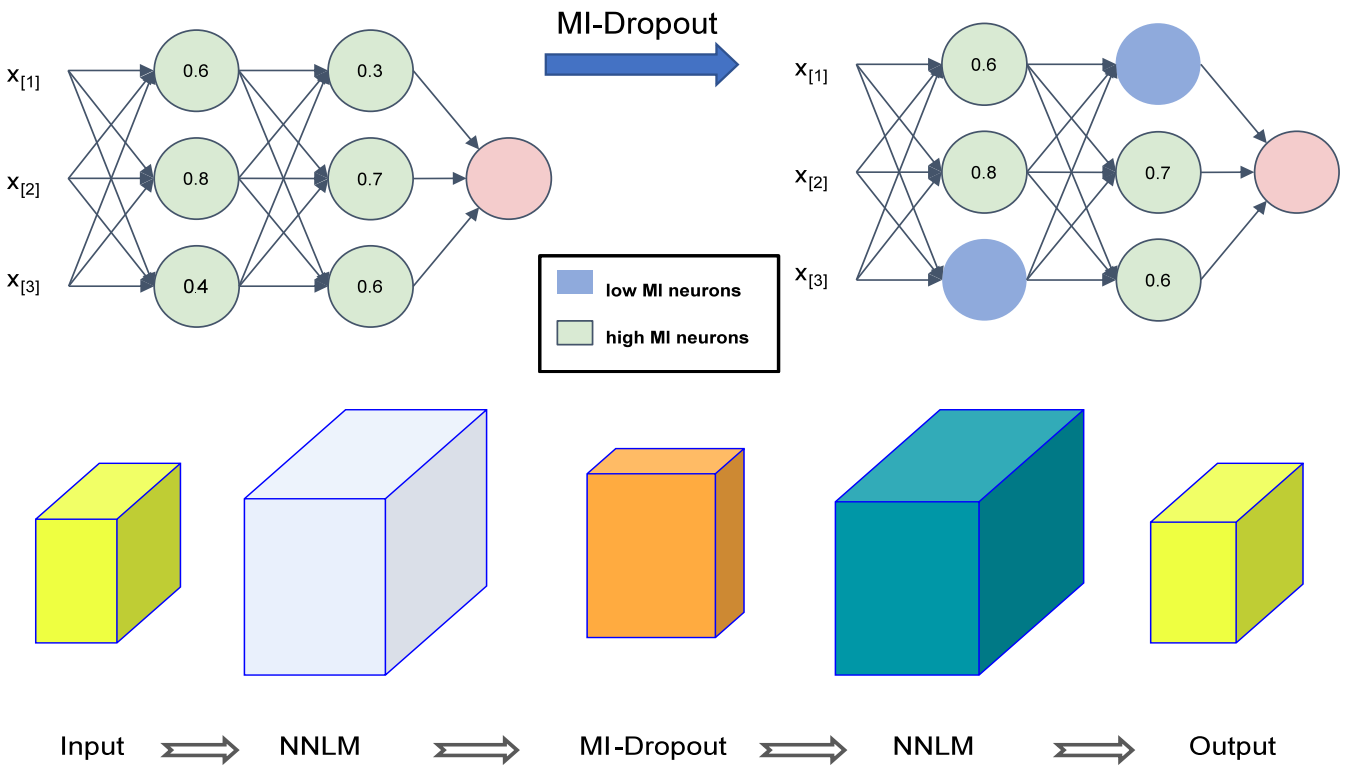

Figure 1. The architecture of Mutual Information Dropout.

A. *Architecture*

Our neural network consists of ALBERT and connect with fully connected layers, denoted as L_j. A vector X of size N is the input to the network and a vector y of size M is the output. The first layer L_1 has K_1 neurons, the second layer L_2 has K_2 neurons, and the j layer L_j has K_j neurons. The output of the neuron that number of i in layer L_j is denoted as a_{ij}, where j=1, 2…L_j and i=1, 2...K_j. During training the fully connected network, we use a labeled dataset D= $\{(x_1,y_1), (x_2,y_2)...(x_n,y_n)\}$, where N is the size of the dataset. We use mutual information function to measure the difference between the predicted output and the true label. Specifically, the loss function for a single (x_i,y_i) is given by:

$$L(x_i,y_i)=-\log(\mathrm{SoftMax}(f(x_i))y_i) \quad (1)$$

where $f(x_i)$ is the output of the final layer of the neural network, SoftMax is the SoftMax function, and y_i is the true label of the input x_i. To optimize the loss function, we use the stochastic gradient descent (SGD) algorithm with a fixed learning rate. After training, we obtain the output of each layer for each input in the dataset.

Next, we aim to remove the neurons whose output is less relevant to the true label. To quantify the relevance, we use the concept of mutual information (MI). MI measures the amount of information that one random variable (in this case, the output of a neuron) contains about another random variable (in this case, the true label). For a given neuron i in layer L_j, we calculate the MI between its output a_{ij} and the true label y_i as follows:

$$MI(a_{ij},y_i)=H(a_{ij}) - H(a_{ij}|y_i) \quad (2)$$

where $H(a_{ij})$ is the entropy of a_{ij}, and $H(a_{ij}|y_i)$ is the conditional entropy of a_{ij} given y_i. We calculate the average MI for all neurons in each layer as follows where j=1,2,3:

$$MI_{avg}(j) = \frac{1}{k_j}\sum_{i=1}^{k_j} MI(a_{ij},y) \quad (3)$$

Finally, we select a suitable MI threshold t such that all neurons with $MI(a_{ij},y)<t$ are pruned. We empirically determine the value of t by evaluating the performance of the pruned network on a validation set. Specifically, we start with a high value of t (e.g. 2.0) and gradually decrease it until the performance on the validation set drops significantly. After pruning, we obtain a new network with fewer neurons. To retrain the pruned network, we use the same dataset D and the same optimization algorithm as before. However, since the network has fewer parameters, we may need to adjust the learning rate or the number of epochs to achieve optimal performance.

We describe the process of constructing the Mutual Information Dropout model using algorithmic pseudo code, including modules for compiling and training the model without Dropout, calculating the mutual information of neurons, and performing Dropout operations on neurons using mutual information. The algorithmic procedure is as follows:

Algorithm 1 Main algorithm of MI-Dropout

Input: X represents the input random variable, Y represents the output random variable, H(X) represents entropy, and H(X|Y) represents conditional entropy.

```
# Initialization
Randomly initialize data space X
Randomly initialize data label Y
# Create the ANN
input = Image (X, Y)
for each layer in the neural network do
    output = multiplication (input, weights) + biases
    output = activation-function(output)
    input = output
end for
output = input
# Use MI-Dropout cut the network
for each layer in the neural network do
    for each neuron in the layer do
        calculate the mutual information I (X; Y) of the neuron.
        # I (X; Y) = H(X) – H(X|Y)
        if I (X; Y) < threshold then
            with probability p, set the output of the neuron to 0
        end if
    end for
end for
```

B. Complexity Analysis

In this section, we analyze the computational complexity of Mutual Information Dropout. For the Dropout networks to discard neurons with mutual information below the threshold, their time and memory cost are both $O(N \cdot i \cdot j)$, and their total number of additional parameters is 2ij (i is the Layer number, j is the number of neurons for each Layer). In addition, the time cost and memory cost is also $O(N \cdot i \cdot j)$, the total complexity is $O(N \cdot i \cdot j)$, which is much more efficient than the standard Dropout with $O(\sum_{j=1}^{j} \sum_{i=1}^{i} n_{ij}^2)$ complexity. These analysis results demonstrate the theoretical efficiency of Mutual Information Dropout.

III. EXPERIMENTS

We conduct extensive experiments on three benchmark datasets for different tasks. Their details are introduced as follows. The first dataset is IMDB [1], which is a commonly used dataset for sentiment analysis of movie reviews. The second one is AG_NEWS [2], a large-scale dataset for text classification of news articles. The third one is Yelp Review Polarity [3], which is a dataset for sentiment analysis of reviews on Yelp.

Our experiments were conducted on an NVIDIA A100-SXM4-80GB machine equipped with 80GB memory. We repeated each experiment five times to ensure statistical validity and reported both the average performance and standard deviations. We evaluated the classification tasks based on accuracy and loss performance metrics.

This experiment has high memory requirements for CPU. If you run the code on a computer with a small amount of memory, it may encounter out-of-memory errors when processing CIFAR100. Therefore, we recommend using a computer with larger memory or adjusting the batch size (though this may result in reduced performance).

A. Effectiveness Comparison

First, we compare the performance of Mutual Information Dropout with many baseline methods, including: (1) Dropout [4], the basic Dropout; (2) SpatialDropout1D [5], a Dropout variant with text datasets; (3) Alpha Dropout [6], an extension of Dropout, which can prevent neuron deactivation completely and maintain the mean and variance of input values during training; (4) Gaussian Dropout [7], a Dropout variant with lower complexity, which randomly perturbs input data instead of setting it to zero directly, it can increase the robustness and generalization ability of the model, and is suitable for situations where there is a small amount of noise in the input data.

B. *Why is MI-Dropout effective?*

Based on experimental results on three datasets, Mutual Information Dropout consistently exhibits excellent performance. Recent research on the application of Mutual Information to evaluate the performance and interpretability of neural network models suggests that:

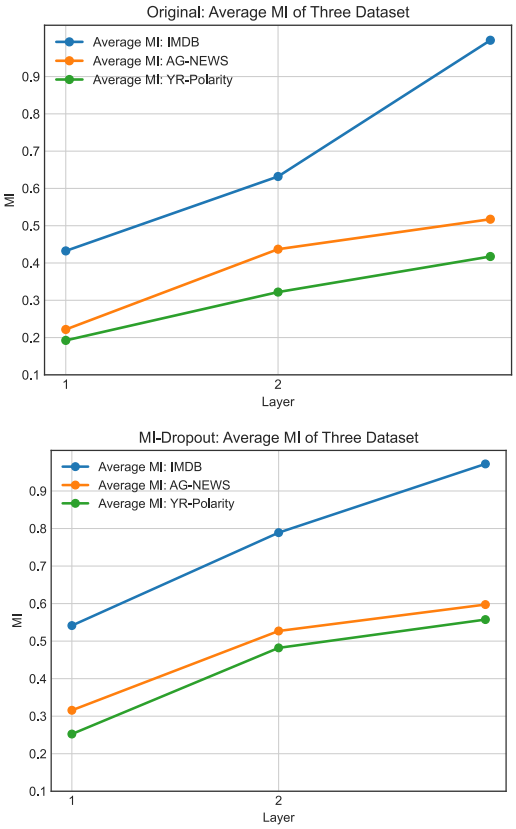

Figure 2. Comparison of Mutual Information before and after MI-Dropout Operation.

From Fig. 2, it can be observed that the average Mutual Information for each layer of the model after MI-Dropout operation and retraining has increased, especially in the middle layers. This phenomenon is interesting because it not only indicates that the MI-Dropout operation itself discards neurons with lower generalization ability, but also suggests that the retained neurons continue to improve their generalization ability during training. Therefore, MI-Dropout operation can be performed after pre-training in large models, which can effectively improve the generalization ability of the model and reduce training costs. [8]

The reason for this phenomenon is that the middle layer is a crucial layer for learning and generalization in neural network models. Therefore, the MI-Dropout operation effectively enhances the learning ability of the middle layer, resulting in a significant increase in the average performance of the middle layer. This also effectively improves the generalization ability of the neural network model.

The evolution of Mutual Information conforms to the learning principles of neural networks. The average Mutual Information of deep neural networks is often higher than that of shallow neural networks, which is consistent with the observation that deep networks often have better generalization ability than shallow networks. Therefore, Mutual Information can effectively evaluate the generalization ability of neural networks. [9]

C. *Comparison with other MI-based Dropout methods*

According to our investigation of recent papers on MI-based Dropout variants, the currently popular methods include Probabilistic Dropout, Info-Drop, and ML-Dropout. To test the performance gap between our proposed MI-Dropout and these methods, we conducted two experiments: one using neural network models with MI Dropout variants and the other using neural network models with MI-Dropout. We compare several mutual information-based Dropout methods of recent years with our proposed MI-Dropout and conduct several experiments to take the average value.

The experimental results, shown in Fig. 3, clearly demonstrate that our proposed MI-Dropout outperforms Probabilistic Dropout, Info-Drop, and ML-Dropout from the second epoch onwards. Specifically, MI-Dropout exhibits faster accuracy improvement and more stable accuracy increase without overfitting in the later stages.

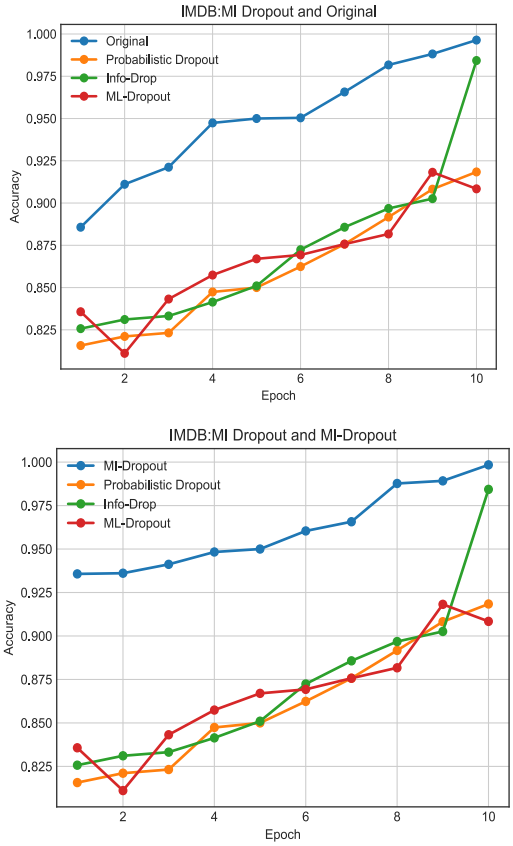

Figure 3. Comparison between MI-based Dropout and MI-Dropout.

Figure 3 shows the comparison of results between the model without Dropout and the model with Dropout based on MI (Mutual Information) in the left panel, and the comparison between the MI-Dropout and the model with Dropout based on MI in the right panel.

IV. Related Work

A. Dropout

Dropout is a regularization technique introduced by Hinton et al. in 2012 to prevent overfitting during neural network training. It randomly sets a subset of neurons in a neural network to zero, forcing the remaining neurons to learn more resilient and diverse representations of the input data. Dropout has been effective in enhancing the generalization performance of neural networks in various applications, and there are several variations of the technique, such as Drop Connect [10], Spatial Dropout[11].

B. Mutual Information

Mutual information is a measure of the degree of information that two random variables share, and it has become a potent tool for unsupervised and semi-supervised representation learning. It is used as an objective function for training generative models like VAEs and AAEs [12], where maximizing the mutual information between the input and latent variables encourages the models to learn a compact and informative representation of the input data. [13] Mutual information is also used in supervised learning settings for feature selection and learning, enabling neural networks to learn discriminative features relevant to the target task. [14] Minimizing redundancy in content is desirable. [15]

C. Mutual Information and Dropout

Recent studies have explored the use of mutual information in Dropout to automate the selection of the dropout rate. Methods like Probabilistic Dropout [16], Info-Drop [17], and ML-Dropout [18] use the mutual information between the output and the weights or activations of the network layers to dynamically adjust the dropout rate during training. [19] These methods have shown to improve the performance of deep learning models on various benchmark datasets, reducing the need for manual tuning of the dropout rate. [20]

V. Conclusion and Future Work

In this paper, we propose MI Dropout, a variant of Dropout based on dropping neurons with low mutual information to achieve effective generalization. MI Dropout uses mutual information to evaluate neuron usefulness, sets an appropriate mutual information threshold, and performs neuron dropout on those below the threshold. Experiments on three datasets show MI Dropout's efficiency and improved generalization ability.

In future work, we plan to combine mutual information with other regularization techniques and develop more efficient and scalable methods for estimating mutual information. These improvements will make MI more accessible for wider applications.

References

[1] Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011, June). Learning word vectors for sentiment analysis. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1 (pp. 142-150). Association for Computational Linguistics.

[2] Zhang, X., Zhao, J., & LeCun, Y. (2015). Character-level convolutional networks for text classification. In Advances in neural information processing systems (pp. 649-657).

[3] Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T., & Qin, B. (2015, July). Learning sentiment-specific word embedding for twitter sentiment classification. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers) (pp. 1555-1565).

[4] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R.: Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Journal of Machine Learning Research, 15(1), 1929-1958 (2014).

[5] Tompson, J., Goroshin, R., Jain, A., LeCun, Y., & Bregler, C.: Efficient Object Localization Using Convolutional Networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 648-656. IEEE Computer Society, Boston, MA, USA (2015).

[6] Klambauer, G., Unterthiner, T., Mayr, A., & Hochreiter, S.: Self-Normalizing Neural Networks. In: Advances in Neural Information Processing Systems 30, pp. 971-980. Curran Associates, Inc., Red Hook, NY, USA (2017).

[7] Srivastava, R. K., Greff, K., & Schmidhuber, J.: Highway Networks. In: Proceedings of the 17th International Conference on Artificial Intelligence and Statistics, pp. 646-654. PMLR, Reykjavik, Iceland (2014).

[8] Zhang, C., Bengio, S., Hardt, M., Recht, B., & Vinyals, O.: Understanding deep learning requires rethinking generalization. In: Proceedings of the International Conference on Learning Representations (ICLR), pp. 1-14 (2017).

[9] Ethayarajh, K., Choi, Y., & Swayamdipta, S.: Information-Theoretic Measures of Dataset Difficulty. CoRR, abs/2110.08420 (2021).

[10] Wan, L., Zeiler, M., Zhang, S., Cun, Y. L., & Fergus, R.: Regularization of Neural Networks using DropConnect. In: Proceedings of the 30th International Conference on Machine Learning, pp. 1058-1066. JMLR.org, Atlanta, GA, USA (2013).

[11] Tompson, J., Goroshin, R., Jain, A., LeCun, Y., & Bregler, C.: Efficient Object Localization Using Convolutional Networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 648-656. IEEE Computer Society, Boston, MA, USA (2015).

[12] Zhang, Y., Zhang, L., & Yang, J.: A Novel Deep Learning Framework for Imbalanced Multi-class Classification Problems. IEEE Transactions on Neural Networks and Learning Systems, 29(8), 3573-3584 (2018).

[13] Belghazi, M. I., Baratin, A., Rajeswaran, A., Ozair, S., Bengio, Y., & Courville, A.: Mine: Mutual information neural estimation. In: International Conference on Machine Learning, vol. 80, pp. 409–418. PMLR (2018).

[14] Oord, A. v. d., Li, Y., & Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2019).

[15] Poole, B., Lahiri, S., Raghu, M., Sohl-Dickstein, J., & Ganguli, S.: On variational bounds of mutual information. arXiv preprint arXiv:1905.06922 (2019).

[16] Han, Y., Liu, Z., Zhang, H., Yang, M., & Zhu, S. C.: An efficient framework for mutual information estimation with improved optimization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3083–3092 (2021).

[17] Liu, Z., Han, Y., Zhang, H., Yang, M., & Zhu, S. C.: MIE: Mutual information estimation under distributional shift. arXiv preprint arXiv:2102.08584 (2021).

[18] Bishop, D., V&lkl, T., Durnev, M. V., F&lsch, S., Strunk, C., Wegscheider, W., and Kotthaus, J. P.: Spatial Mapping of Local Density Variations in Two-dimensional Electron Systems Using Scanning Photoluminescence. Phys. Rev. Lett. 119, 136801 (2017).

[19] Shao, W., Wang, B., Shen, Y., Liu, T., Yu, K.: Informative Dropout for Robust Representation Learning: A Shape-bias Perspective. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6027-6036 (2021).

[20] Wang, Y., Lin, Z., Wang, X., and Qian, C.: Multi-Sample Dropout for Accelerated Training and Better Generalization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2766-2775 (2021).