A fully connected network topology with Neural Network

Sif .K. Ebis[1]

[1]Wassit Education Directorate, Ministry of Education ,Iraq, Email: saifkather@gmail.com

Received: 13.04.2024 Revised : 18.05.2024 Accepted: 25.05.2024

ABSTRACT
We propose a novel "Completely Connected Networks" deep network structure to enhance the model's discriminability for small patches within the receptive field (CCN) , The conventional convolutional layer uses linear filters to examine the input followed by a nonlinear activation function Instead we build more complex miniature neural networks to pool information from the receptive field The tiny neural network is instantiated as a multilayer perceptron a robust function approximator Micro net- works much like CNN are slid over the input to produce follow-up feature images, after which transmitted to the next layer for further processing , The architecture allows for the stacking of multiple instances to realise deep CNN , The micro network's enhanced local modelling allows us to employ categorization layer feature map pooling on a world scale which both increases interpretability and decreases the likelihood of overfitting in comparison to more traditional fully connected layers. We demonstrated that CNN yields state-of-the-art categorization results on the CIFAR-10 and CIFAR-100 datasets as well as reasonably good results on the SVHN and MNIST datasets.

Keywords: Completely Connected Networks (CCN), Neural Network (CNN),MLP conv layer , P2P Network, Client–server Networks.

1. INTRODUCTION

To use a neural network convolutional has layers of both neural and pooling neurons (CNNs)[1] Diagrams dubbed "feature maps" result when a linear filter is applied to the input of a convolution layer , the affected by natural is used alongside the linear filter's interior component and finally a nonlinear activation function is applied to each local area of the input.

In this paper we contend that CNN's generalised linear model convolution filter (GLM)is not very complex , We say that a property is generic if it applies to all manifestations of the same idea Substituting a more resilient nonlinear function approximator for the GLM can increase the local model's generalisation ability When the concept variants fall neatly opposite the GLM-specified dividing line is able to efficiently abstract from the data[2] , As a result traditional CNN assumes incorrectly that hidden ideas can be partitioned along a linear axis , Because of this models that represent these concepts are typically very non- linear effects of the input, despite the fact that the data for a similar concept may be linear frequently resides on a nonlinear surface The "micro network" structure used in CNN approximates nonlinear functions and it takes the place of the generalised linear model (GLM) in statistical analysis In this research, we build a multilayer perceptron a type of neural network that can be trained via back-propagation and is thus suitable for use as a micro network.

Figure 1 depicts a comparison between the CNN and our finished "MLP conv layer" architectures [3], All convolutional layers including linear conv and multi-layer perceptron conv function by encoding the local receptive field into a feature vector , The MLP conv uses a multilayer perceptron (MLP) with several completely connected layers and nonlinear activation functions to map the input local patch to the output feature vector this MLP is shared by all of the receptor classes , the feature maps are obtained by sliding the MLP.In Figure 1 we can see the differences between the MLP conv layer and the linear conv layer.The linear convolution layer employs a linear filter, and the MLP conv layer's micro network is in charge of fine-tuning the output (In this study, we opted to use a layered perceptron) Like convolutional neural networks both layers use the input to determine how confident they can be in the hidden idea (CNNs) Multi-layer perceptron (MLP) conv is used extensively throughout the CCN's construction Specifically[4] , the idea of "Completely Connected Networks" refers to the reality that the deep network's MLP conv layers comprise micro networks (MLP) that are essential to the network as a whole (CCN) We use a global average pooling layer to generate a confidence vector for each class and then feed that vector into a softmax layer in place of the completely connected layers used by a standard CNN In traditional CNN, the category-level information as from target cost layer is not transparently transmitted back to the preceding convolution layer due to the fully linked levels that operate as a black box[5].Conversely the micro network's improved local modelling enables more useful and interpretable global average pooling by requiring congruence between feature maps and categories In contrast to the fully connected layers which rely largely on dropout regularisation to prevent overfitting global average pooling is a structural regularizer that reliably and automatically safeguards against overfitting of the entire network.

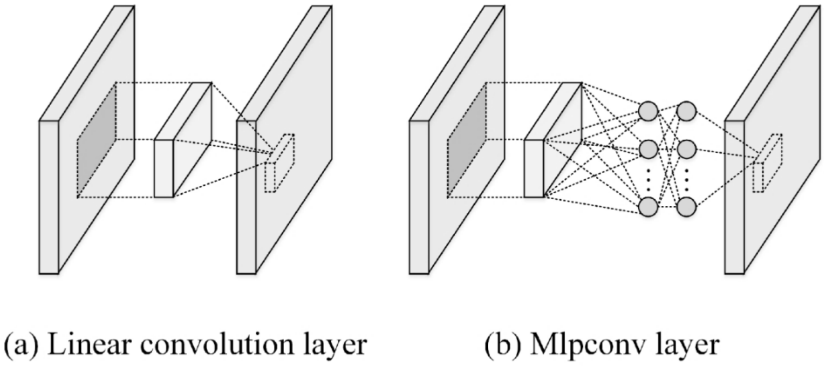

Figure 1. Linear convolution vs. MLP conv.

1.1. Mlpconv Layer
The feature vector is the input mapping in the local awareness field of view of the mlpconv layer which has a linear convolutional layer and an MLP[6], the mlpconv layer uses multiple fully connected layers with nonlinear activation functions to extract completely connected network topology target feature information transform it into a feature map and use the feature map as the input of the next layer[7].

2. Networks of Neural Convolutions
Features maps are created to start with linear convolution layers and then move on to nonlinear activation functions in traditional convolutional CNN which are composed of layered levels of neural and spatial pooling When examples of the hidden ideas can be linearly , separated linear convolution is adequate for abstraction[8] , However models that accomplish effective abstraction are typically extremely non-linear functions of the incoming data In conventional CNN[9], person linear frames can be taught to identify variants of the same concept but having too many filters for a single concept increases the load on the subsequent layer Therefore it is helpful to perform a higher degree of generalisation on each local fix before merging them into more generalised ideas By maximising pooling over affine feature maps, the new maxout , network lowers the number of feature maps making it a piecewise linear approximator that can approximate any convex functions.
This enhancement gives the network top-tier capabilities across a variety of test data sets In more complicated cases where the ranges of the hidden ideas are more dispersed a more general function approximator would be required , To this end an innovative "Network In Network" architecture [10], is suggested where a Mini-networks are inserted into each convolution operation to compute granular features for specific areas. The concept of a Numerous works have suggested a nano network that is shifted over the input[11], however these networks are all either issue or have only one layer.

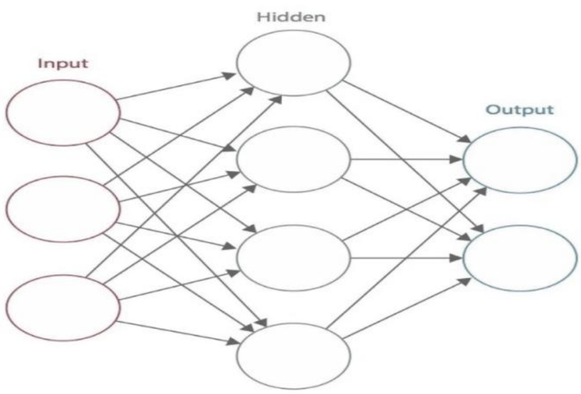

Figure 2. artificial Networks of Neural Convolutions

2.1. Engineering of Computer Networks

The physical and conceptual planning of the data-transmission software, hardware, protocols[12], and media is known as computer network architecture, to put it simply computer architecture describes the way in which programmes are laid out and jobs are assigned to machines[13].
There are two common network architectures:

2.1.1. Peer-to-Peer Network

In a peer-to-peer network, all the machines are equal participants in the network and share equally in the handling of data[14].

- A group of no more than about 10 machines can benefit from a peer-to-peer network.
- A peer-to-peer system does not rely on a central computer.

Each computer has its own set of privileges for accessing the shared resources which can cause issues if the computer housing the resource goes offline.

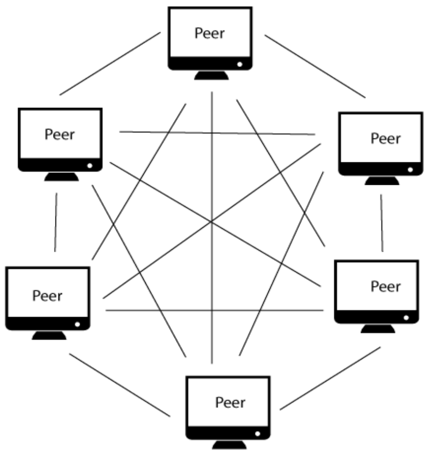

Figure 3. P2P Networks model

2.1.1.1. The Benefits of a P2P Network

- Since there is no specialised computer involved, the price is significantly reduced.
- All other machines will continue to function normally even if one of them fails.
- Since each machine is responsible for its own upkeep, it requires little effort to set up and keep running smoothly.

2.1.1.2. There are some drawbacks to using a P2P network, such as the absence of an organised system.

- Since the material is unique in each place, it can't be backed up.
- Inasmuch as the gadget is handled independently, there is a security risk.

2.1.2. Computer Network with Clients and Servers

Users, or "clients," in a client/server network paradigm retrieve data like media files and other content stored on a centralised machine, or "Server."
A server is the primary computer in a network[15], while the other devices are known as clients.
All the heavy lifting, such as managing the network and ensuring security, is done by the server.
A server is the central hub that controls access to and storage of data and peripherals like printers, scanners, and file folders.
A server mediates the interactions between all the customers. If Client1 wishes to transmit data to Client2[16], it must first ask the server for approval. The server then responds to the first client with the second client's information.

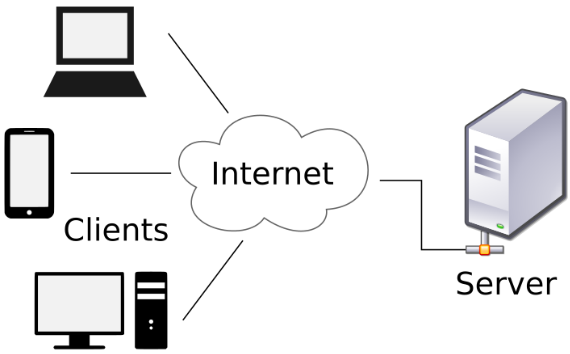

Figure 4. Client–serverNetworks model

2.1.2.1. Client/Server networks have many benefits[17].

- The consolidated infrastructure is hosted on a Client/Server network. So now it's simple to create backups of the information.
- A central server boosts the efficiency of a Client/Server network.
- Due to centralization in server management, data is safer in a client/server network.
- It speeds up the process of pooling resources as well.

2.1.2.2. Negatives of a Client/Server system:

- There is a high cost associated with client/server networks because of the need for a powerful server with lots of RAM.
- The Network Operating System (NOS) on a computer is what makes its tools available to customers, but NOS comes at a hefty price.
- A full-time network controller is essential for overseeing all the system's tools.

3. Levels of Convolutional Neural Networks in MLPs

Because it can approximate more complex models of the latent concepts, a An optimal method for feature extraction is a universal function approximator of the local regions when no priors about the distributions of the latent concepts are available. There are a few well-known universal function approximators[18], including the radial basis network and the multilayer perceptron. In this study, we select layered perceptron for two main factors. To begin, the back-propagation method of training convolutional neural networks is consistent with the multilayer perceptron structure. The second is that multilayer perceptron can be a deep model, which is in line with the principle of feature reuse[19]. In this study, MLP is used instead of GLM to convolve over the input, and this new layer is referred to as MLP conv. Linear convolutional layers and MLP conv layers are depicted differently in Figure 1. Here is an illustration of the MLP conv layer's computation.

Multilayer perceptrons have n levels, and so this expression uses that value. In a layered perceptron, the activation function is a rectified linear unit, From the perspective of cross channel (cross feature map) pooling, CCN can be thought of as being equal to cascading cross channel parametric pooling on a regular convolution layer[20]. The incoming feature maps undergo weighted linear merging at each pooling layer before being passed through a rectifier linear unit[21]. In subsequent levels, the previously cross-channel pooled feature images are pooled again. This hierarchical data-sharing framework facilitates complicated and re-learnable cross-channel exchanges via parameterized pools.

3.1. Connectivity and Networks

Network connectivity is another type of metric used to examine the quality of the connections between various nodes in the network. The term "network topology" is associated with this subject[22]. it describes the overall make-up and layout of the network.

Hub, linear, tree, and star topologies are just a few of the many possible configurations for a network. Each of these configurations takes a slightly different approach to establishing a network through which electronic gadgets can talk to one another[23]. Each type of network connection comes with its own set of benefits and drawbacks.

When discussing the growing variety of networks and the interconnections between them, IT professionals, especially network administrators and network analysts, often refer to connectivity as if it were a piece of the networking puzzle.

Ad hoc networks and car networks, to name just two examples, are two instances of the new kinds of networks that operate based on different kinds of communication models[24]. Network administrators and support personnel are responsible for more than just keeping things talking; they must also make security a top priority. This is because data security is intrinsically linked to the reliability of networking infrastructures.

4. Experiments And Results

4.1. Review

4.2. We use CNN on the CIFAR-10, CIFAR-100, SVHN, and MNIST databases to accomplish this After the MLP convolution layer a spatial pooling layer performs a downsampling operation on a ratio of two from the original raw picture in every single iteration Every one of the dataset-specific networks is an MLP conv tri-layer architecture. Dropout a regularizer is applied to the outputs of every MLP conv layer except the final one Instead of having fully connected levels at the network's centre[25], all the examples in the experiments part use global average pooling.

4.3. In preprocessing , the datasets are separated into training and testing collections Initially , the learning rates and weights are set appropriately by hand During network training 96-member[26] , minibatch sizes are used. Up until that point, the starting weights and exercise speeds will be maintained is no more improvement in efficiency on the training dataset at which time the training data is lowered by a factor of 10. Only once through the procedure will you reach a training set of 2% of the initial population.

4.4. CIFAR-10

There are a total of 59,000 training pictures and 9,000 assessment images available in the CIFAR-10 collection, which is divided evenly among 10 categories of natural images[27]. Each picture is a 32x32 RGB photograph. The dataset is processed using the same global contrast normalisation and ZCA bleaching that Goodfellow et al. implemented in the maxout network. As confirmation data, we use the last 9,000 pictures from the training collection.

In this exercise, we match the amount of feature maps in each MLP conv layer to that of the associated maxout network[28]. The validation collection is used to fine-tune two hyper-parameters: the area of the local receptive field and the rate of weight decline. Next, we reset the network's hyper-parameters and re-train it is using that includes the initial training set as well as the confirmation collection. The final product is a prototype replica. On this dataset, we achieve a test error of 10.41%, a gain of over 2% over the state-of-the-art. Table 1 displays the results of a comparison with earlier approaches.

Table 1. shows the findings of a study that compared the current method to its predecessors

Method	Test Error
Stochastic Pooling	15.13%
CNN+Spearmint	14.98%
Conv.maxout+Dropout	11.68%
CCN+ Dropout	10.41%
CNN+Spearmint+Data Augmentation	9.50%
Conv.maxout+Dropout+Data Augmentation	9.38%
DropConnect+12networks+Data Augmentation	9.32%

Our experiments show that implementing dropout between CCN's MLP conv layers has a positive effect on the network's performance by enhancing the model's generalizability. Incorporating dropout layers in-between the MLP conv layers resulted in a more than 20% reduction in test error[29], also came to this conclusion, so it must be true. Accordingly, abandonment is an additional factor.

4.5. CIFAR-100

CIFAR-100 is a companion collection to CIFAR-10 in terms of quantity and organisation, but it includes 100 additional institutions. groups instead of 10. Therefore there are only ten times as many images in the CIFAR-10 collection overall[30]. Rather than fine-tune the hyper-parameters for CIFAR-100, we stick with the same parameters we used for CIFAR-10. The very last MLP conv layer only differs in that it generates 100 feature maps. For CIFAR-100, we achieve a test error of 37.68 percent, which is better than the state-of-the-art performance without data supplementation by over one percentage point. Table 2 provides a detailed breakdown of the score comparison.

Table 2. Percentage of incorrect answers on the CIFAR-100 test set.

Strategy	Faulty Testing
Successfully Mastered the Art of Pool	43.71%
Random Sampling	42.51%
Comparing the maximum and minimum values Plus	38.57%
The Use of Priors Based on Forest Structures	36.85%

4.5.1. Identification Codes for Homes in Google Maps' Street View

There are a total of 630,420 pictures in the SVHN dataset, all of which are 32x32 colours and are split between a collection of data used for training and another set used for evaluation, and an additional set. Classifying the central digit in each picture is the goal of this dataset. serve as the basis for the training and testing process. For confirmation, we use 400 samples drawn from the training set for each class and 200 samples drawn from the additional set for each class. The remaining portions of both the primary and secondary exercise sets are utilised. It is never done in practise to train a model on the validation set; instead, it is only used to guide hyper-parameter selection.

The dataset was preprocessed using the same local contrast normalisation that was Like CIFAR-10, SVHN employs a three-layer MLP conv architecture followed by a global-average pooling layer. Our results as shown in Table 3.

Table 3. Variations in SVHN mistake rates on test sets.

Strategy	Faulty Testing
Random Sampling	2.80%
Amplifier, voltage regulator, current limiter	2.78%
Combining a Rectifier with a Dropout and a Mechanical	2.68%
Comparing the maximum and minimum values Plus dropout	2.47%
CCNPlusDropout	2.35%
Recognizing Larger Numbers	2.16%

4.6. MNIST

The MNIST dataset features 0-9 penned by hand in a 28x28 grid. All told, there are 60,000 images used for training and 10,000 for assessment[31] ,The same kind of network architecture as in CIFAR-10 is used for this data collection However fewer feature images are produced because of each MLP conv layer Compared to CIFAR-10, MNIST is a less complex dataset so fewer factors can be used, This dataset serves as a testbed for our approach without the need for supplemental data Table 4 displays the results alongside a comparison to works by other authors that also used neural structures.

Table 4. Error rates on the MNIST test set for different approaches

Method	Test Error
2-Layer CNN + 2-Layer NN	0.53%
Stochastic Pooling	0.47%
CCN + Dropout	0.49%
Conv. maxout + Dropout	0.45%

5. Regularization by Globally Pooling Averages

Both the completely connected layer and the global average pooling layer use linear transformations of the vectorized feature maps to achieve their respective results , The key is in the change matrix , The array of transformations is prepended to be used as a benchmark for comparison across the world and it is only non-zero on components of the block diagonal that have the same value Back-propagation optimization is applied to the values of fully connected layers' transformation vectors which can be quite large , The regularisation impact of global average pooling is investigated by swapping out Generally speaking, a completely linked network's one while keeping the rest of the model constant , This model was tested whether or not there was a quitter stage preceding the completely associated linear layer, The CIFAR-10 dataset is used to evaluate both models and the results are compared in Table 5.

Table 5. The global pooling mean is compared to the completely connected layer.

Method	Test Error
mlpconv + Fully Connected	12.1%
mlpconv + Fully Connected + Dropout	13.3%
mlpconv + Global Average Pooling	9.1%

Table 5 shows that the worst performance (12.1%) came from a fully linked layer without dropout regularisation This is to be expected since the fully linked layer tends to overfit the training data in the absence of a regularizer There was a noticeable difference in error rates between both the three testing methods[32], with average global pooling getting the lowest failure rate (9.1%) by including a single hidden layer before the fully connected one. Following this, we investigate whether normal CNNs gain the same regularisation benefits from global average pooling as deep neural networks. We use a three-layer convolutional neural network (CNN) with a single local link layer as described in Given that the typical global pooling method only permits a single feature map per group, we decrease the extracted features of the target line segment from 16 to 10resulting in a fully connected layer with dropout[33]. To create a network with the same properties as one that uses dropout Plus without the latter, we can simply replace its completely connected top layer with the CIFAR-10 dataset was used as a standard for comparison in all analyses.

6. CONCLUSIONS

We proposed a novel deep network architecture for classification that we refer to as "Completely Connected Networks" (CNN) The conventional CNN's completely connected layers are replaced in this novel architecture by MLP conv layers which convolve the input and layer with the aid of multilayer perceptrons Together MLP conv layers and global average pooling which acts as an internal regularizer to avoid global overfitting make for a powerful model, We demonstrated that the two pillars of CCN obtain assert results just on CIFAR-10, CIFAR-100, and SVHN datasets. Showing Indicator Maps allowed us to demonstrate that the extracted features produced by the final MLP conv layer of CCN were trust classification diagrams lending credence to the concept of employing CCN for object identification.

REFERENCES

[1] L. Zhang and S. Hiroyuki, “Automated test input generation for convolutional neural networks by implementing multi-objective evolutionary algorithms,” in 2020 Eighth International Symposium on Computing and Networking Workshops (CANDARW), 2020, pp. 157–163.

[2] Y. Bengio, A. Courville, and P. Vincent, “Representation learning: A review and new perspectives,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no. 8, pp. 1798–1828, 2013.

[3] R. Aziz, C. K. Verma, and N. Srivastava, “Artificial neural network classification of high dimensional data with novel optimization approach of dimension reduction,” Ann. Data Sci., vol. 5, pp. 615–635, 2018.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” Commun. ACM, vol. 60, no. 6, pp. 84–90, 2017.

[5] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, “Improving neural networks by preventing co-adaptation of feature detectors,” arXiv Prepr. arXiv1207.0580, 2012.

[6] H. Alaeddine and M. Jihene, “Deep network in network,” Neural Comput. Appl., vol. 33, pp. 1453–1465, 2021.

[7] K. B. Obaid, S. Zeebaree, and O. M. Ahmed, “Deep learning models based on image classification: a review,” Int. J. Sci. Bus., vol. 4, no. 11, pp. 75–81, 2020.

[8] F. Gama, E. Isufi, G. Leus, and A. Ribeiro, “Graphs, convolutions, and neural networks: From graph filters to graph neural networks,” IEEE Signal Process. Mag., vol. 37, no. 6, pp. 128–138, 2020.

[9] B. Yang, G. Bender, Q. V Le, and J. Ngiam, “Condconv: Conditionally parameterized convolutions for efficient inference,” Adv. Neural Inf. Process. Syst., vol. 32, 2019.

[10] J. I.-Z. Chen and K.-L. Lai, “Deep convolution neural network model for credit-card fraud detection and alert,” J. Artif. Intell., vol. 3, no. 02, pp. 101–112, 2021.

[11] S. Alonso-Monsalve, A. L. Suárez-Cetrulo, A. Cervantes, and D. Quintana, “Convolution on neural networks for high-frequency trend prediction of cryptocurrency exchange rates using technical indicators,” Expert Syst. Appl., vol. 149, p. 113250, 2020.

[12] Y. Zhang, X. Lan, J. Ren, and L. Cai, “Efficient computing resource sharing for mobile edge-cloud computing networks,” IEEE/ACM Trans. Netw., vol. 28, no. 3, pp. 1227–1240, 2020.

[13] Y. Wang, X. Tao, X. Zhang, P. Zhang, and Y. T. Hou, “Cooperative task offloading in three-tier mobile

computing networks: An ADMM framework," IEEE Trans. Veh. Technol., vol. 68, no. 3, pp. 2763–2776, 2019.
[14] J. Zhang, H. Guo, J. Liu, and Y. Zhang, "Task offloading in vehicular edge computing networks: A load-balancing solution," IEEE Trans. Veh. Technol., vol. 69, no. 2, pp. 2092–2104, 2019.
[15] Q. Luo, C. Li, T. H. Luan, and W. Shi, "EdgeVCD: Intelligent algorithm-inspired content distribution in vehicular edge computing network," IEEE Internet things J., vol. 7, no. 6, pp. 5562–5579, 2020.
[16] J. West, Data Communication and Computer Networks: A Business User's Approach. Cengage Learning, 2022.
[17] L. Zhang and N. Ansari, "Latency-aware IoT service provisioning in UAV-aided mobile-edge computing networks," IEEE Internet Things J., vol. 7, no. 10, pp. 10573–10580, 2020.
[18] M. Desai and M. Shah, "An anatomization on breast cancer detection and diagnosis employing multi-layer perceptron neural network (MLP) and Convolutional neural network (CNN)," Clin. eHealth, vol. 4, pp. 1–11, 2021.
[19] S. T. Ikram et al., "Anomaly detection using XGBoost ensemble of deep neural network models," Cybern. Inf. Technol., vol. 21, no. 3, pp. 175–188, 2021.
[20] J.-H. Wang, G.-F. Lin, M.-J. Chang, I.-H. Huang, and Y.-R. Chen, "Real-time water-level forecasting using dilated causal convolutional neural networks," Water Resour. Manag., vol. 33, pp. 3759–3780, 2019.
[21] A. Kumar and N. Sachdeva, "Multi-input integrative learning using deep neural networks and transfer learning for cyberbullying detection in real-time code-mix data," Multimed. Syst., vol. 28, no. 6, pp. 2027–2041, 2022.
[22] C. Haythornthwaite, "Learning, connectivity and networks," Inf. Learn. Sci., vol. 120, no. 1/2, pp. 19–38, 2019.
[23] Z. Fodor, A. Horváth, Z. Hidasi, A. A. Gouw, C. J. Stam, and G. Csukly, "EEG alpha and beta band functional connectivity and network structure mark hub overload in mild cognitive impairment during memory maintenance," Front. Aging Neurosci., vol. 13, p. 680200, 2021.
[24] J. R. Foerster, X. Costa-Perez, and R. V. Prasad, "Communications for iot: Connectivity and Networking," IEEE Internet Things Mag., vol. 3, no. 1, pp. 6–7, 2020.
[25] O. K. Oyedotun, D. Aouada, and B. Ottersten, "Improved highway network block for training very deep neural networks," IEEE Access, vol. 8, pp. 176758–176773, 2020.
[26] T. Nguyen, R. Novak, L. Xiao, and J. Lee, "Dataset distillation with infinitely wide convolutional networks," Adv. Neural Inf. Process. Syst., vol. 34, pp. 5186–5198, 2021.
[27] M. Singh, A. Sinha, N. Kumari, H. Machiraju, B. Krishnamurthy, and V. N. Balasubramanian, "Harnessing the vulnerability of latent layers in adversarially trained models," arXiv Prepr. arXiv1905.05186, 2019.
[28] S. Mohseni, M. Pitale, J. B. S. Yadawa, and Z. Wang, "Self-supervised learning for generalizable out-of-distribution detection," in Proceedings of the AAAI Conference on Artificial Intelligence, 2020, vol. 34, no. 04, pp. 5216–5223.
[29] K. Takayama, I. Sato, T. Suzuki, R. Kawakami, K. Uto, and K. Shinoda, "Smooth transfer learning for source-to-target generalization," in NeurIPS 2021 Workshop on Distribution Shifts: Connecting Methods and Applications, 2021.
[30] J. Li, A. S. Rakin, X. Chen, Z. He, D. Fan, and C. Chakrabarti, "Ressfl: A resistance transfer framework for defending model inversion attack in split federated learning," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 10194–10202.
[31] D. Lee and Y. Cheon, "Soft labeling affects out-of-distribution detection of deep neural networks," arXiv Prepr. arXiv2007.03212, 2020.
[32] Z. Liu, C. Cao, F. Tao, Y. Li, and X. Lin, "From Spatial to Spectral Domain, a New Perspective for Detecting Adversarial Examples," Secur. Commun. Networks, vol. 2022, 2022.
[33] Q. Tian, K. Kuang, K. Jiang, F. Wu, and Y. Wang, "Analysis and applications of class-wise robustness in adversarial training," in Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, 2021, pp. 1561–1570.