iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 **Issue:** VII **Month of publication:** July 2021

DOI: https://doi.org/10.22214/ijraset.2021.36499

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VII July 2021- Available at www.ijraset.com

Dropout- A Detailed Survey

Sakshi S Lad[1], Poonam Bhagat[2]
[1]Undergraduate Student, [2]Assistant Professor, School of Computer Engineering & Technology, MIT-WPU, India

Abstract: Deep Neural Networks are very complex and have large number of parameters. Shortlisting the parameters that influence the model prediction is not possible as each has equal significance. These neural nets have powerful learning skills can model training data well enough. However, in most of these conditions, the models are over-fitting. Combining predictions from large neural nets where neurons are co-dependent alters the performance of the model. Dropout addresses the problem of overfitting and slow convergence in deep neural nets. The core concept of dropout technique is to randomly drop units and their connections from the neural network during training phase. This prevents units from co-adapting and thus improving the performance. The central mechanism behind dropout is to take a large model that overfits easily and repeatedly sample and train smaller sub-models from it. This paper provides an introduction to dropout, the history behind its design and various dropout methods.
Keywords: dropout methods, neural network, overfitting

I. INTRODUCTION

Deep Neural Networks have evolved into powerful predictive models with remarkable performance and are used in many fields. The whole idea behind Neural Nets is to mimic how the human brain works. Deep Learning is a sub-branch of machine learning inspired by the structure of human brain. Deep Learning Algorithms try to mimic how the human brain works. It attempts follow the learning process of humans to draw conclusions. It trains fully connected Neural network with large amount of neurons which makes the model prone to problem of overfitting. Couple of these neurons get over familiar with the training data that they even learn and memorize the noise in the data. When too many neurons extract the same features, it adds more significance to those features thereby resulting in duplicate feature extraction and wasting computer resources. There are a lot of techniques to address the problem of overfitting that includes early stopping the training before the model learns noises, training the model with more data, data augmentation, feature selection, regularization techniques but Dropout works the best.

Dropout was introduced in 2012 as a technique to avoid overfitting in the paper "Improving neural networks by preventing co-adaptation of feature detectors". The term "dropout" refers to dropping few random units along with all its incoming and outgoing connections. Certain units are ignored during the training phase of randomnly selected neurons. Each neuron has a dropout rate p assigned to it. The probability of each neuron being removed from the network is p. It is not completely removed. It still has a probability of $(1-p)$ of getting included in the next training step. Initially, each neuron in the neural network with probability 0.5 was omitted in each training iteration and all neurons were included during testing. This technique drastically improved results on a variety of datasets.

A. Dropout

Inspired by the original dropout method, a wide range of dropout methods have been proposed to use in neural architectures. The most recommended is Standard dropout which suggests dropout probability p=0.2 on the input layer and a probability p=0.5 on the hidden layers. The application of dropout isn't only limited to overcoming the problem of overfitting but also compressing the neural nets to produce on output using less computational resources. Different dropouts methods can be applied to different neural architectures. For instance, SpatialDropout, Cutout, CorrDrop, DropBlock provide better results only when applied to CNN. Dropout will decrease the rate of convergence but will generally result in better model. This paper gives an introduction to dropout methods.

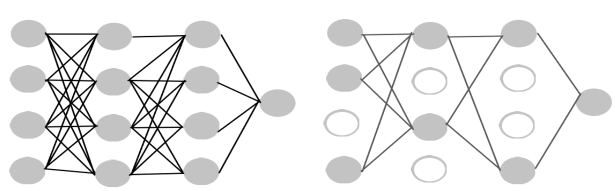


Fig.1 Dropout Model

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VII July 2021- Available at www.ijraset.com

II. BACKGROUND

In Neural Network, a fully connected layer occupies most of the parameters due to which neurons become co-dependent during the training. This curbs the individual power of each neuron leading to overfitting in the model. The motivation behind the design for dropout comes from the role of sex in evolution. There are two types of reproduction, namely, sexual and asexual. In sexual reproduction offspring is produced by adding equal amount of genes from each parent and a very small amount of random mutation while in asexual reproduction a slightly mutated copy of the parent's genes does the work. It is specious that asexual reproduction should be better as good set of genes are passed on directly to the offspring but those co-adapt too much to sustain and the law of natural selection states the survival of the fittest. On the other hand, sexual reproduction produces robust individuals by reducing complex co-adaptions in sets of genes. It is the way most advanced living organisms evolve. Similarly, after applying a dropout layer in the model each hidden unit learns to work with a randomly chosen sample of other units to make the model robust.

A. Dropout Techniques

A lot of dropout methods have been proposed since 2012 and used in different architectures. In a fully connected network, each layer except the outer is given a probability *p*. The gist of the important ones are given below.

1) *Standard Dropout:* The first dropout method was Standard dropout proposed by Hinton et al. It is the most widely used dropout. It suggests a probability $p = 0.2$ on the input layer and probability $p = 0.5$ on the hidden layers. In the training phase at every iteration, each neuron has a probability *p* of getting omitted from the network but it is not completed removed. It still has a probability $p = (1\text{-}p)$ of getting included in the next time. This probability of omission for each neuron is based a Bernoulli distribution. No dropout is applied in testing phase.

Mathematically, standard dropout during training for a neural network layer is given by:

$$y = f(Wx) \circ m, \; m_i \sim \text{Bernoulli}(1 - p)$$

where y is the layer output, $f(\cdot)$ is the activation function, W is the layer weight matrix, x is the layer input, and m is the layer dropout, with each element m_i being 0 with probability p.

Once trained, the layer output is given by:

$$y = (1 - p)f(Wx).$$

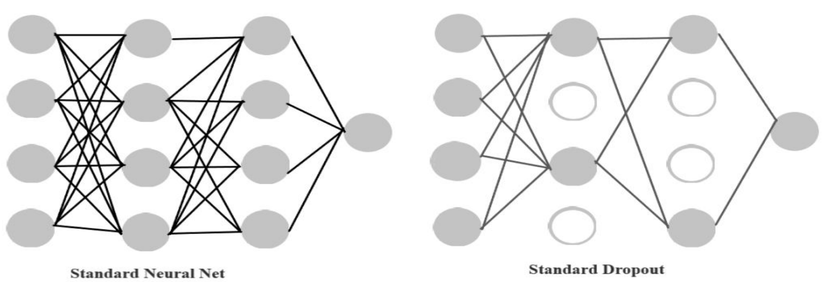


Fig. 2 Standard Dropout

2) *Standout dropout :* This technique was proposed by L Ba et al. It suggests adaptive probability *p* of omission based on the value of the weights in neural network. It is also based on Bernoulli distribution and can be applied to feature-learning models and CNN architecture.
3) *DropConnect:* The DropConnect dropout was proposed by Wan et al. It suggests applying dropout to the weights and bias linking the neurons in the neural network. It is based on Gaussian approximation which also outputs the model uncertainty.

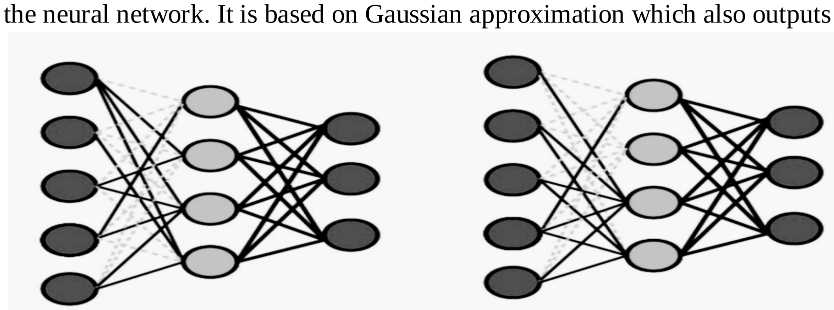


Fig. 3 DropConnect

4) *Annealed Dropout:* The Annealed dropout which was proposed by Rennie et al. suggests randomly dropping out a fixed percentage of neurons in each training iteration for improving independency of individual neuron.
5) *Bounded Random Dropout*: The Bounded random dropout was proposed by Buyck et al. It suggests adding noise at the start of the training and decreasing it to 0 towards the end.
6) *RNNDrop*: The RNNDrop dropout was proposed with Moon et al which is specifically used for RNN architecture. It suggests to randomly generate a Bernoulli dropout mask for a sequence and using the same throughout the sequence. By this, it learns temporary dependencies avoiding the co-adaption.
7) *Monte Carlo:* The Monte Carlo dropout was proposed by Gal et al. Generally, dropout is applied at training phase but it is suggested apply Monte Carlo dropout during the testing phase after the model is already trained using a standard dropout. That way multiple predictions are generates at each instance. An actual prediction is made after averaging them. Monte Carlo dropout not only attends to the problem of overfitting but also in getting model uncertainty.
8) *Spatial Dropout:* The Spatial dropout was proposed by Tompson et al especially for CNN architecture. It suggests using dropout mask per feature map in the training phase with an omission probability *p*. The entire feature map is dropped and not included during the pooling.

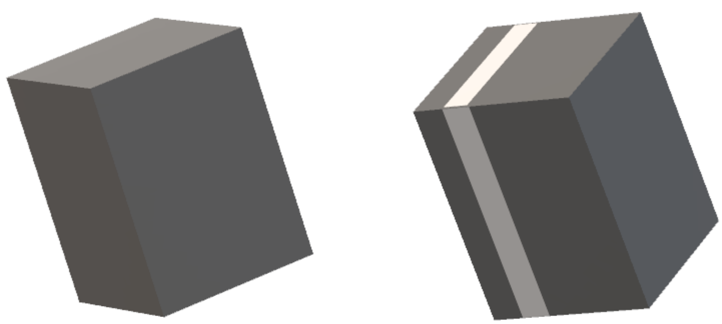


Fig 4. Dropout applied per feature map

9) *Cutout Dropout:* The Cutout dropout was proposed for CNN architecture mostly in image recognition field by DeVries et al. It suggests applying dropout to few areas of feature map so that architecture can recognize less obvious features too but generating too many images is computationally expensive and slows down the training process.

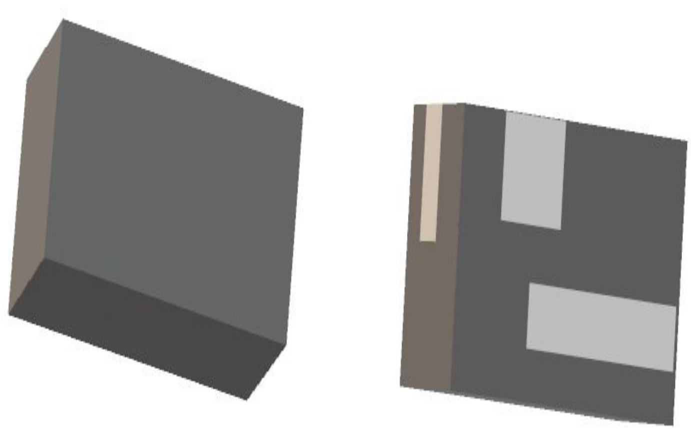


Fig 5. Dropout applied on few areas

10) *Deterministic Dropout:* The Deterministic dropout was proposed by B.Santra et al which suggests dropping the unimportant connections while still ensuring propagation of class discriminative information. These randomly dropped connections are identified using random forest algorithm. It works great with smaller datasets as well as noisy datasets.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VII July 2021- Available at www.ijraset.com

11) *Excitation Dropout*- The Excitation dropout was proposed by Zunino et al. It suggests dropping out those neurons which contribute more to the decision making in the training phase which basically means dropping with higher probability, the most relevant neurons which is exactly opposite to the Adaptive dropout strategy. This forces the model to learn more based on limited features available.

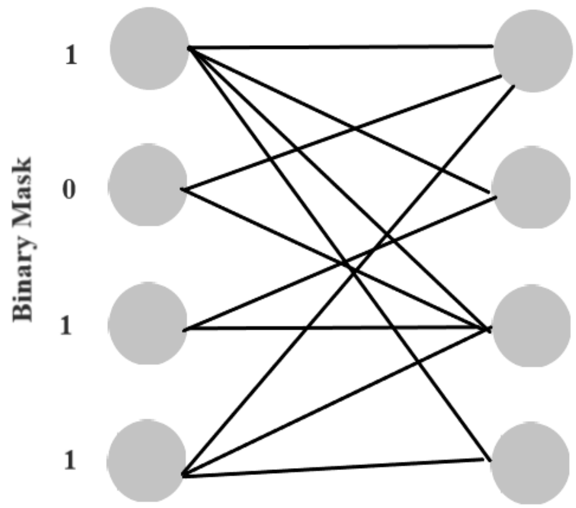


Fig.6 Excitation Dropout applied using Binary Mask

Year 2021	Year 2020	Year 2019	Year 2018	Year 2017
Excitation Dropout(A Zunino et al)	CorrDrop(Yuyuan Zeng et al)	Spectral dropout (Khan et al)	Targeted dropout (Gomez et al)	Cutout (DeVries et al)
Deterministic Dropout(B Santra et al)	DropGrad(HY Tseng et al)	Ising dropout (Salehinejad et al)	Information dropout (Achille et al)	Dropout for model compression (Neklyudov et al)
Edropout(H Salehinejad et al)	Automatic Dropout(V Dodballapur et al)	Adversarial dropout for RNNs (Park et al)	Adversarial dropout methods (Park et alVb Saito et al)	Variational dropout for sparsification (Molchanov et al)
		Effective dropout for CNNs (Cai et al)	DropBlock(G Ghiasi et al)	Concrete dropout (Gal et al)
		Weighted Channel Dropout (Hou et al)	\	Curriculum dropout (Morerio et al)
		Test-Time Dropout(Cortés-Ciriano et al)		AlphaTdivergence dropout in Bayesian NNs (Li et al)
		Batch DropBlock(Z Dai et al)		WeightTdropped LSTMs (Merity et al)
				Fraternal dropout (Żołna et al)

Table 1. Dropout methodologies proposed from Year 2021 to Year 2017

Year 2016	Year 2015	Year 2014	Year 2013	Year 2012
Stochastic depth (Huang et al)	Variational dropout (Kingma et al)	Multiplicative Gaussian noise (Srivastava et al)	Analysis of dropout as averaging (Baldi et al)	Standard dropout (Hinton et al)
Evolutionary dropout (Li et al)	RNNDrop (Moon et al)	Annealed dropout (Rennie et al)	Standout (Ba et al)	
Variational RNN dropout (Gal et al)	MaxTpooling dropout (Wu et al)	Empirical analysis of dropout (WardeTFarley et al)	Fast dropout (Wang et al)	
Hidden state update dropout (Semeniuta et al)	Spatial dropout (Tompson et al)	Bounded random dropout(Duyck et al)	Dropconnect (Wan et al)	
Monte Carlo dropout (Gal et al)	Dropout for RNNs (Zaremba et al)		Maxout (Goodfellow et al)	
Selective CNN dropout (Park et al)	Dropout as data augmentation (Bouthillier et al)		Dropout as weight regularization (Wager et al)	
Swapout (Singh et al)				

Table 2. Dropout methodologies proposed from Year 2016 to Year 2012

III. COMPARISON

Though dropout is the most effective technique to deal with overfitting in the model, it can alter the performance in some cases. Applying dropout right before the last layer doesn't give promising results beczuse then the network doesn't have the ability to correct the errors. A smaller network has low capacity. Applying dropout layer will lower the capacity more and hiurt the model performance. Therefore, a smaller neural network doesn't require a dropout layer. A dropout layer should only be applied whe the model is going to train until convergence because lower training time may give worse results. Thus, applying dropout layer everytime won't give good accuracy and performance. A complete knowledge of the architecture is must. Dropout may sometimes cause the problem of underfitting the model as it randomly sets activations to zero. The recommended dropout rate for input layer is 0.1 and for hidden layers is 0.5 and 0.8 for better performance of the model. Training the model on large data and then applying dropout will also help to achieve a better performance.

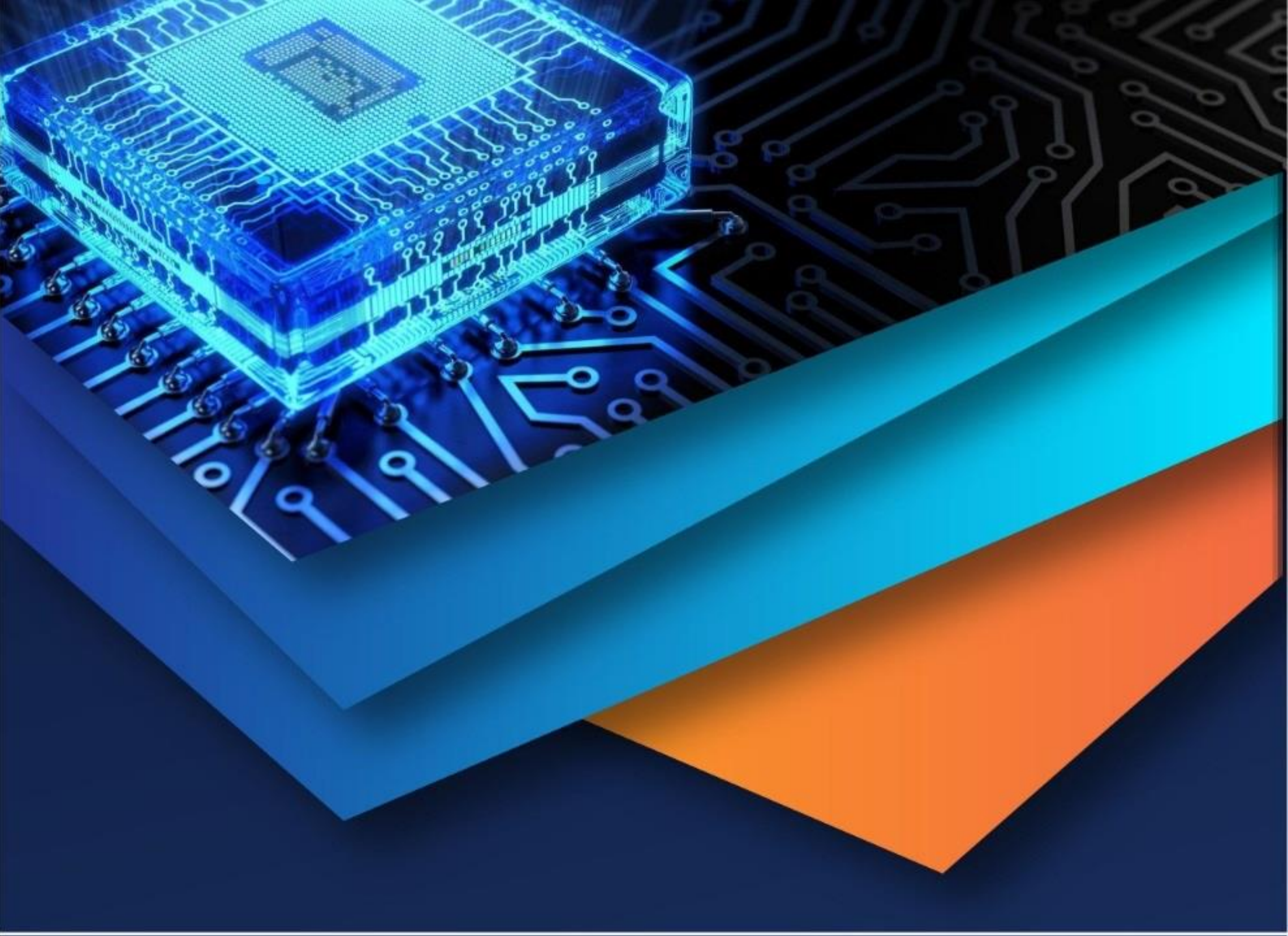
IV. CONCLUSIONS

Dropout methods increases the performance of neural network and reduces the overfitting in them when used correctly. A minor co-adaption may be good for the model but it does not do well with unseen data. Temporarily inactivating random neurons and removing them on the forward pass and not updating their weights on the backward pass may not only solve the overfitting problem but will also make efficient use of computation resources. Dropouts like Monte Carlo can also be used to measure the model uncertainty. Dropout methods have shown their utility and potential for 10 years now and still new techniques influenced by Bayesian, Gaussian are under research. There's one thing that isn't still quite clearly explained that why a dropout rate of 0.1 for input layer and 0.5 or 0.8 for hidden layers gives better regularization and results. But the cost we pay when we use dropout layer in the model is the slow rate on inference.

REFERENCES

[1] Srivastava, Nitish, et al. "Dropout: a simple way to prevent neural networks from overfitting." The journal of machine learning research 15.1 (2014): 1929-1958.
[2] Portela, R. D. S., et al. "Lung Region Segmentation in Chest X-Ray Images using Deep Convolutional Neural Networks." 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC). IEEE, 2020.
[3] Ruan, Xinru, Cheng Tian, and Wei Xiang. "Research on Face Recognition Based on Improved Dropout Algorithm." 2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC). IEEE, 2020.
[4] Labach, Alex, Hojjat Salehinejad, and Shahrokh Valaee. "Survey of dropout methods for deep neural networks." arXiv preprint arXiv:1904.13310 (2019).
[5] Wan, Li, et al. "Regularization of neural networks using dropconnect." International conference on machine learning. PMLR, 2013.
[6] Miok, Kristian, et al. "Generating data using Monte Carlo dropout." 2019 IEEE 15th International Conference on Intelligent Computer Communication and Processing (ICCP). IEEE, 2019.

[7] Srivastava, Nitish. "Improving neural networks with dropout." University of Toronto 182.566 (2013): 7.
[8] Wu, Haibing, and Xiaodong Gu. "Max-pooling dropout for regularization of convolutional neural networks." International Conference on Neural Information Processing. Springer, Cham, 2015.
[9] Gomez, Aidan N., et al. "Learning sparse networks using targeted dropout." arXiv preprint arXiv:1905.13678 (2019).
[10] Gal, Yarin, and Zoubin Ghahramani. "Dropout as a bayesian approximation: Representing model uncertainty in deep learning." international conference on machine learning. PMLR, 2016
[11] Labach, Alex, Hojjat Salehinejad, and Shahrokh Valaee. "Survey of dropout methods for deep neural networks." arXiv preprint arXiv:1904.13310 (2019).
[12] Baldi, Pierre, and Peter J. Sadowski. "Understanding dropout." Advances in neural information processing systems 26 (2013): 2814-2822.
[13] Duyck, James, Min Hyung Lee, and Eric Lei. "Modified dropout for training neural network." Paper Published online (2014).
[14] Moon, Taesup, et al. "Rnndrop: A novel dropout for rnns in asr." 2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU). IEEE, 2015.



10.22214/IJRASET

INDEX COPERNICUS 45.98

IMPACT FACTOR: 7.129

TOGETHER WE REACH THE GOAL

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY