Vol. 06, No. 4 (2024) 1889-1896, doi: 10.24874/PES.SI.25.03A.022

Proceedings on Engineering Sciences

www.pesjournal.net

DEEP NON-NEGATIVE MATRIX FACTORIZATION MODEL FOR CLUSTERING-BASED IMAGE DENOISING

Shaily Malik[1]
Nishtha Jatana
Divya Jatain
Geetanjali Sharma
Vaani Garg
Amita Yadav

Received 06.01.2024.
Received in revised form 15.02.2024.
Accepted 20.03.2024.
UDC – 004.932

Keywords:

Clustering, denoising, autoencoders, homogeneity, dimensionality reduction, quantization, non-negative matrix factorization.

A B S T R A C T

Technologies like self-driving cars and cleaning robots are emerging as mainstream technologies. These technologies make use of cognitive recognition. Non-negative matrix factorization (NMF) is one such technique that is popularly used for computer vision and hidden pattern recognition. NMF is prone to noises because it assumes the image signal to be linearly reconstructed. This work proposes an algorithm to increase the effectiveness of NMF and reduces the data to lower dimensions and add informational presentation which improves the clustering results of NMF.

The effectiveness of the proposed model is measured by comparing them on attributes namely accuracy, homogeneity, and inertia. Some of the models that we used include K-means, PCA+K-means, NMF+K-means, Autoencoder + PCA + K-means. Our proposed model is observed to be the most effective for clustering denoised data. The algorithm also takes care of the different fault detections and gives a non-linear method based on NMF. Here, we first used autoencoders which are given input data to learn the non-linear mapping so that it can be transformed into high-dimensional space. By using the decomposition rule, we divided our feature space into two parts: The first one comprises the encoder, NMF, and decoder. This method of DNMF is a non-linear framework that can further be extended to other linear methods. The proposed method also expands the NMF's application range as it can also accept non-negative input.

[1] Corresponding author: Shaily Malik
Email: shaily.singh99@gmail.com

1. INTRODUCTION

In the past few years due to the recent increase in demand for intelligent automobiles, a lot of research developments are taking place to further extemporize self-driving vehicles. Different domains like computer vision, audio analysis, loss data generation, image segmentation, hidden pattern detection, and capturing information from raw data require advanced algorithms and the most advanced architectures. There is an extensive amount of research needed to find efficient algorithms and architectures that give promising results. If we take an example of a car image, any given part loss can easily be identified (suppose the front wheels are missing), and we can easily manipulate the car image by attaching that missing part. To solve such interpretability issues, we propose DNMF (Deep Non-negative matrix factorization) that can help to learn deep representation which contains spatial and temporal information of data. To discover the underlying potential of NMF with the deep neural network, we have found a way to connect NMF and autoencoders. Encoders convert input data to low dimensional data, while decoders do exactly the opposite that is transform the encoded data back to original data by reconstitution of encoded data. This way helps to accomplish better accuracy. But these existing techniques are dependent on a two-stage way to add NMF and encoders-decoders. Autoencoders help to extract deep representation so that NMF can work on top of that. Since both of these parts are mutually divided the above method does not learn from each other. Additionally, the above approach is sensitive to noise in input data which negatively impacts the reliability and performance in a real-world application. (Duong, Hseih, Bao & Wang, 2014) To resolve such problems, we propose a DNMF (Deep Non-Negative Matrix Factorization) model that contains multiple layers of NMF in combination with a pooling layer followed by back-propagation. In DNMF architecture, two networks learn from each other, on one side we have a student network, and on another, we have a supervisor network. This model learns a deep representation of data and delivers results with noised data. The supervisor and student network combination handle interpretability loss as it is given in the trained parameter. In our model, we have applied three types of losses Symmetric loss which supervises the supervisor network, non-negative constraint loss for handling interpretability loss and apposition loss which supervises the student network.

2. RELATED WORK

Non-negative matrix factorization (NMF) is a prominent algorithm that is well-established for an application like computer vision, data clustering, signal processing, and bioinformatics (Duong et al,2014)(Ye, Chen & Zheng, 2018)(Buciu Nikoliadis & Pitas, 2008)(Jia, Liu, Hou & Kwong,2021)(Gocken & Yaktubay, 2019). To extract useful information from such massive data is like finding a needle in a stack of hay. Algorithms like Linear Embedding, Principal component analysis are not independently capable to operate on such an amount of data effectively and efficiently. To handle such a vast amount of data, there is a need for algorithms that can effectively do dimensionality reduction. In NMF we have to represent a data matrix which is the multiplication result of the base matrix and weight matrix. In NMF, we want to decompose the matrix into two matrices. For a given matrix V with m rows and n columns, where each element is non-negative, NMF will decompose the matrix W with m rows and r columns and matrix H with r rows and n columns where each of their elements is also non-negative (Jia et al,2021).

In machine learning, it is often necessary to reduce the feature space of a dataset, for ease of computation. NMF is a relatively new way of reducing the dimensionality of our dataset, into a linear combination of bases. Existing methods include Principal Component Analysis (PCA) which weights with positive and negative values to blend one representation. Vector Quantization (VQ), which is sort of the nearest neighbor algorithm, consists of bases of prototype observation and the single closest one is selected (Bando, Mimua, Itoyama, Yoshi & Kawahara, 2018). NMF accepts non-negative constraints and therefore, it can deal with the representation of non-negative data features in a better way. It is similar to PCA, wherein it assigns weight to a set of bases to blend a representative observation, but the weights are bound to be positive, which is how part-based representation is learned by NMF. A customized k-means clustering-based picture segmentation technique is suggested. This technique's modified form incorporates the k-means clustering algorithm, a de-noising factor connected to each pixel's velocity field, and edge distinction using the Canny edge detector (Islam, Nahar, Islam, Islam, Mukhajee & Ali, 2021). The retrieval of celebrity cartoon images is a difficult task because there are variances in terms of caricatures and styles. built a Clustering-based Tree with backtracking to present a fresh way for effectively retrieving cartoon images.

3. METHODOLOGY

This section elaborates on the data used, the model used, the optimizations needed, the algorithmic complexity, and the evaluation and scoring method used in this work.

3.1. Dataset

The dataset plays an impeccable role in determining the kind of feature extraction and eventual retrieval efficiency. The image and text retrieval task generally require challenging natural language processing and a convolutional neural network model for extracting the features in a condensed form. Thus, two major datasets with diverse categories were used. Firstly, we used 70,000 labeled gray images describing the handwritten digits from 0 to 9, from the MNIST dataset, which are dimension 28×28 in size. This work carries out the

DNMF model process and other comparative methods, and measures the quality by the coincidence with the correct labels, using the images in the datasets. It is one of the most popular datasets commonly found being exploited in domains like object detection, segmentation, captioning datasets, and natural language processing.

Secondly, the Fashion-MNIST dataset consisting of a similarly big corpus of 70,000, 28x28 dimensional gray images with 10 different labels was also employed. It has a vivid variety of images of fashion clothing items such as trousers, coats, sneakers, and many others.

3.2. The Deep Non-Negative Matrix Factorization Model

The deep matrix factorization model comprises two sections, first the supervisor network, and second the student network. Both networks contain encoders and decoders. The contrast between the student network and the supervisor network is that the student network contains an NMF package between encoders and decoders. As we all know deep neural network (Bhattamishra, 2018) (Guo, Zhao, Nie, Ruan & Li, 2020)(Salakhutdinov & Murray, 2008) is good at taking out deep representations in unsupervised learning by remodeling the inputs. The main objective of the deep neural network is analogous to the supervisor network in DNMF.

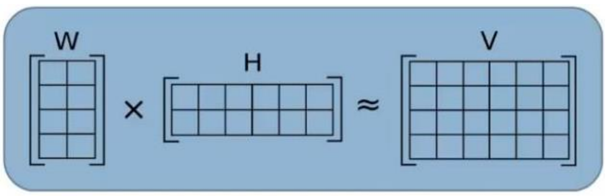

Figure 1. Matrix V is decomposed into a low-dimensionality matrix W and a matrix H.

The user can specify r, the inner dimension of W and H, as long as it is r < min(m,n). We see that each column of V, v can be calculated as v = W*h.

In a similar fashion to the supervisor network, encoders and decoders pull out deep representations of given data. Some amount of noise and attenuation is added to add similarity in real-world data. Both networks learn from each other. By mixing the noise with the data, the student network is given noisy data, to generate noise-free data. Whereas, supervisor data get noiseless data. In our DNMF architecture, we have inserted an NMF module between encoders and decoders which discloses interpretability (Dhand, Sheoran, Agarwal & Biswas, 2022). The work of NMF is to factorize the output of encoders and with the help of matrix multiplication helps to decode the generated representation. (Sangwan & Bhatnagar, 2020) If we take the example of vanilla NMF, we can say that it adheres to matrix factorization patterns and non-negative constraints. In our model, we do not do two-stage training but in the student network, we have secured the NMF between encoders and decoders. For installing the hidden feature, we have used the sigmoid function in the NMF module.

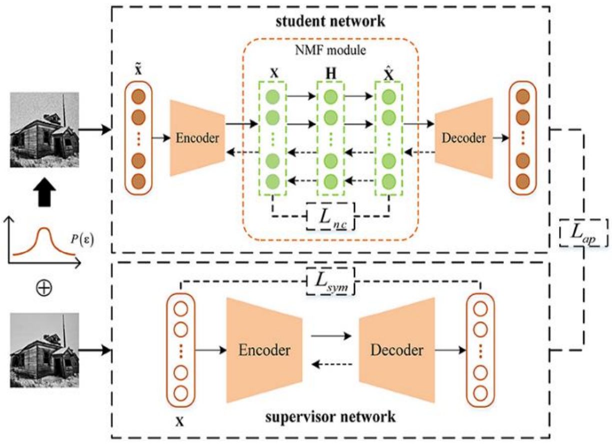

Figure 2. Architecture of the DNMF model consisting of supervisor and student network

From the architecture of DNMF, it is visible that we are feeding our data to both a supervised network and a student network. We are transforming the data to retrieve the hidden pattern and information in the supervisor network. By adding random noise as input to the student network, we have tried a real work-like scenario where we get lots of adulterated input data. NMF with encoders and decoders helped in the process to generate feature extraction and reconstructing the output. NMF has proved its efficiency in varied areas (Mao & Saul, 2004)(Devarajan,2006)(Magkanas, Bagan, Sistac & Garcia, 2021) (Kherwa & Bansal,2019). We can retrieve deeply hidden features with the help of both networks. Student networks' main objective is to learn features without disturbance whereas supervisor networks assist in finding untapped information.

3.3. Optimizations

The optimization of the model primarily deals with improved loss-generating methodology. It involves a combination of three losses, namely, the symmetric loss, non-negative constraint loss, and apposition loss, whose aggregation forms the interpretability loss function. All of these losses contribute in their way to reconstructing the original data from the noisy data. The symmetric loss is the typical one that contributes to the reconstruction loss for generating the original image. Taking the symmetric decoder and encoder into account, the symmetric loss can be represented as:

$$L_{sym} = \sum_{i=1}^{N} \|\mathbf{x}_i - \mathbf{y}_i\|_2^2 \qquad (1)$$

Here, the ||·||2 denotes the Euclidean norm and the number of data points is represented by N. The ith original data and the ith reconstructed data are expressed as xi and yi respectively.

This symmetric loss is the guiding force for the supervisor network to draw out the essential features from the reconstructed information.

Similarly, the student network is dependent on the apposition loss to gather the appositive features and it contains two additional layers as compared to the number of layers in the supervisor network. This loss can be expressed as:

$$L_{ap} = \sum_{i=1}^{N} \left(\sum_{l=1}^{L/2} \left\| \mathbf{h}_i^{(l)} - \tilde{\mathbf{h}}_i^{(l)} \right\|_2^2 + \sum_{l=L/2+1}^{L} \left\| \mathbf{h}_i^{(l)} - \tilde{\mathbf{h}}_i^{(l+2)} \right\|_2^2 \right) \quad (2)$$

Here, the i[th] input data is xi.xi=xi(0), and the corresponding extracted feature from the lth layer of the student network is denoted by hi.

Then there is another kind of loss that occurs due to the difference between the input to the decoder and the output of the encoder which is passed on to the NMF network, called the non-negative constraint loss. This occurs when the supervisor network hands over all the extracted features and hidden patterns in the data to the student network which in turn extracts the relevant information from the noise that would contribute to the network's intrinsic learning's. It is expressed as:

$$L_{nc} = \sum_{i=1}^{N} \left\| \mathbf{X}_i - \hat{\mathbf{X}}_i \right\|_F^2 = \sum_{i=1}^{N} \left\| \mathbf{X}_i - \mathrm{ReLU}(\mathbf{W}\mathbf{H}_i) \right\|_F^2 = \sum_{i=1}^{N} \left\| \mathbf{X}_i - \mathrm{ReLU}\big(\mathbf{W}\sigma(\mathbf{W}'\mathbf{X}_i)\big) \right\|_F^2 \quad (3)$$

The aggregation of a weighted contribution of these three losses is the interpretability loss function and it can be represented as:

$$L_{in} = \alpha L_{sym} + \beta L_{ap} + \gamma L_{nc} \quad (4)$$

Symmetric functions produce the same loss when they underestimate and overestimate the same absolute error. However, an asymmetric stall function applies a different penalty to different stall directions.

Non-negative constraints: Every decision variable in any linear programming model must be positive whether the objective function is to maximize or minimize the net present value of an asset.

The lack of interpretability (the ability to explain or present in terms that a human can understand) and the introduction of potential bias has given rise to ethical and legal problems. Here, the weightage of all three losses, symmetric loss, non-negative constraint loss, and apposition loss is fine-tuned by the hyperparameters α, β, and γ.

3.4. Algorithm Complexity

Let's say, there are N samples present in each of the input datasets and the dataset containing noise. Then the number of iterations performed, assuming that after processing the entire dataset n times the overall algorithm involved in the method converges, is of the order of O(nN).

There are two major networks involved in the model that perform differently, but important tasks, and each of them takes part in processing the datasets in each iteration. One of them is the student network that comprises a fixed-size supplementary NMF module layer and the other one is a supervisor network with 'S' number of layers, thus causing the DNMF to have a depth of the order of O(S). The changes in parameters and extraction of features involved in the whole process are a result of the forward feed and back-propagation performed on each of the layers. Suppose there are a maximum of M features in all the layers in the model, then the space complexity and the number of parameters in each layer will be O(M2). The time complexity of the forward feed and the back-propagation also needs to be accounted for to get the overall complexity. So, the complexity of the addition and multiplication operations performed on the parameters during each of the processes of forward feed and back-propagation is O(M2). The associated complexity of the activation function and the derivation function is O(M). Hence, the time complexity of each layer in the model is of the order of O(M2).

Thus, algorithm complexity for the overall model can be boiled down to the order of O(nNSM2) because the time complexity for each symmetric loss, non-negative constraint loss, and apposition loss are O(M), which is insignificant in comparison to the overall complexity. Also, the space complexity for the model is of the order of O(SM2).

3.5. Evaluation and Scoring

The performance metrics like accuracy, homogeneity, and inertia were used for assessing and evaluating the performance of the denoising clustering models (Shivaprasad, Guru, Kavitha & Saritha, 2022)(Gao, Shen, Yu & Zhang, 2020). Clustering is usually employed to solve unsupervised learning problems, that can be differentiated into different segments and this process heavily relies on the number of clusters. Therefore, making the correct number of clusters can deeply affect the efficiency as well as allocation of resources. This process is usually done before we do a heavy amount of computational process. Therefore, the amount of reliance on this process is very essential for further computational accuracy.

For the comparative analysis of different methods, we utilized the accuracy metric, which is generally used to measure the characteristics of the classification. In the proposed model, accuracy represents the estimate of how well the clustering has been done based on the class labels and is defined as:

$$\text{accuracy}(y, \hat{y}) = \max_{\text{perm}} \frac{1}{n} \sum_{i=0}^{n-1} 1(\text{perm}(\hat{y}_i) = y_i) \tag{5}$$

where P is the set of all permutations in [1: K] where K is the number of clusters.

Homogeneity is characterized by the homogeneous nature of an image or object. Homogeneous clustering refers to clustering where all data points in each cluster belong to only a particular class. The score for homogeneity can vary between 0 (least homogeneous) to 1 (maximal homogeneity). It's defined as:

$$h = 1 - \frac{H(Y_{\text{true}} | Y_{\text{pred}})}{H(Y_{\text{true}})} \tag{6}$$

Another valuable metric, inertia, is an indicator of the quality of clustering by K-means. It is defined as the sum of squares of the distance between the centroid of the cluster and each data point in it.

$$\sum_{i=1}^{N} (x_i - C_k)^2 \tag{7}$$

A model with good clustering capability is marked by the qualities of a low number of clusters and a low value of inertia. Therefore, these metrics need to be evaluated for each model used. Finally, we can compare the results procured by applying this process to the two datasets.

4. RESULTS AND DISCUSSION

Accuracy, Homogeneity, and Inertia are performance measures that quantify the quality of clustering of the images concerning the query. These metrics are applied to the clusters obtained by applying different models in the MNIST and fashion-MNIST datasets.

Table1. Comparative analysis of models on the MNIST dataset

Comparative Methods	**Accuracy**	**Homogeneity**	**Inertia**
K-means	0.5726	0.4816	2359030.8549
PCA + K-means	0.5238	0.419	490183.8788
NMF + K-means	0.4414	0.3649	658.0024
AE + PCA + K-means	0.5531	0.4687	3673136.0
AE + NMF + K-means	0.3817	0.2754	1351.0283
DNMF	0.5922	0.5737	5484.9726

In the above table1 we can observe the accuracy of the machine learning techniques K-means :0.5726, PCA + K-means: 0.5238, NMF + K-means:0.4414, AE + PCA + K-means: 0.5531, AE + NMF + K-means: 0.3817, DNMF:0.5922. Here we got K-means technique 0.5726 accuracy.

In this work, we have discussed how different models perform clustering on denoised images. It shows the accuracy, homogeneity, and inertia of clustering across two datasets, the first being MNIST (70k images) and the second fashion MNIST (70k images). The accuracy of DNMF for MNIST (text to image) is 0.5922 and for fashion-MNIST is 0.5013. and if we observe the Homogeneity and Inertia K-means is acceptable. Upon comparing the results across both datasets, we can say that DNMF has better clustering capability for noisy datasets.

Table 2. Comparative analysis of different clustering models on the fashion-MNIST dataset

Comparative Methods	Accuracy	Homogeneity	Inertia
K-means	0.4765	0.4081	3368350.4198
PCA + K-means	0.4474	0.3469	371902.0
NMF + K-means	0.3986	0.2927	545.6482
AE + PCA + K-means	0.4618	0.3824	5232518.1045
AE + NMF + K-means	0.3722	0.2170	1583.8133
DNMF	0.5013	0.4652	3496.0538

5. CONCLUSION AND FUTURE SCOPE

This paper proposes a model that recognizes the deep interpretable representations of denoising information. This DNMF model broadly comprises two essential networks, the first one being a supervisor network that deals with the noiseless data, and the other one being a student network that works on the non-negative constraints and deals with retrieving relevant information from data containing noise. It serves the purpose of producing part-based representations from this noisy data. Extracting the vital meaning or aspects from this kind of data is the responsibility of the interpretability loss in the model. Comparing the results obtained from the proposed DNMF and other models, by applying them to the two datasets, shows the effectiveness and better performance of the DNMF model. The model becomes more efficient as more samples are fed and the quality of results should go up. Also, more deep matrix factorization models employing similar architecture can be explored for representing multimodal data.

References:

Bando, Y., Mimura, M., Itoyama, K., Yoshii, K., & Kawahara, T. (2018, April). Statistical speech enhancement based on probabilistic integration of variational autoencoder and non-negative matrix factorization. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 716-720). IEEE. doi: 10.1109/ICASSP.2018.8461530

Bhattamishra, S. (2018). Deep probabilistic nmf using denoising autoencoders. *International Journal of Machine Learning and Computing*, *8*(1), 49-53. doi: 10.18178/ijmlc.2018.8.1.662

Buciu, I., Nikolaidis, N., & Pitas, I. (2008). Nonnegative matrix factorization in polynomial feature space. *IEEE Transactions on Neural Networks*, *19*(6), 1090-1100. doi: 10.1109/TNN.2008.2000162

Devarajan, K. (2006, August). Nonnegative matrix factorization—A new paradigm for large-scale biological data analysis. In *Proceedings of the Joint Statistical Meetings. Joint Statistical Meetings* (pp. 6-10). doi:10.1371/journal.pcbi.1000029

Dhand, G., Sheoran, K., Agarwal, P., & Biswas, S. S. (2022). Deep enriched salp swarm optimization based bidirectional-long short term memory model for healthcare monitoring system in big data. *Informatics in Medicine Unlocked*, *32*, 101010. doi:10.1016/j.imu.2022.101010

Duong, V. H., Hsieh, W. C., Bao, P. T., & Wang, J. C. (2014, September). An overview of kernel based nonnegative matrix factorization. In *2014 International Conference on Orange Technologies* (pp. 227-231). IEEE. doi: 10.1109/ICOT.2014.6956641

Gao, H. N., Shen, D. H., Yu, L., & Zhang, W. C. (2020). Identification of cutting chatter through deep learning and classification. *International Journal of Simulation Modelling*, *19*(4), 667–677. https://doi.org/10.2507/IJSIMM19-4-CO16

Gocken, T., & Yaktubay, M. (2019). Comparison of different clustering algorithms via genetic algorithm for vrptw. *International Journal of Simulation Modelling*, *18*(4), 574–585. https://doi.org/10.2507/IJSIMM18(4)485

Guo, Y., Zhou, D., Nie, R., Ruan, X., & Li, W. (2020). DeepANF: A deep attentive neural framework with distributed representation for chromatin accessibility prediction. *Neurocomputing*, *379*, 305-318. doi: 10.1016/j.neucom.2019.10.091

Islam, M. Z., Nahar, S., Islam, S. S., Islam, S., Mukherjee, A., & Ali, L. E. (2021, September). Customized K-Means Clustering Based Color Image Segmentation Measuring PRI. In *2021 International Conference on Electronics, Communications and Information Technology (ICECIT)* (pp. 1-4). IEEE. doi.org/10.1109/ICECIT54077.2021.9641094

Jia, Y., Liu, H., Hou, J., & Kwong, S. (2020). Semisupervised adaptive symmetric non-negative matrix factorization. *IEEE transactions on cybernetics*, *51*(5), 2550-2562. doi: 10.1109/TCYB.2020.2969684

Kherwa, P., & Bansal, P. (2019). Topic modeling: a comprehensive review. *EAI Endorsed transactions on scalable information systems*, *7*(24). doi: http://dx.doi.org/10.4108/eai.13-7-2018.159623

Magkanas, G., Bagán, H., Sistach, M. C., & García, J. F. (2021). Illuminated manuscript analysis methodology using MA-XRF and NMF: Application on the Liber Feudorum Maior. *Microchemical Journal*, *165*, 106112. doi: 10.1016/j.microc.2021.106112.

Mao, Y., & Saul, L. K. (2004, October). Modeling distances in large-scale networks by matrix factorization. In *Proceedings of the 4th ACM SIGCOMM conference on Internet Measurement* (pp. 278-287). doi:10.1145/1028788.1028827

Salakhutdinov, R., & Murray, I. (2008, July). On the quantitative analysis of deep belief networks. In *Proceedings of the 25th international conference on Machine learning* (pp. 872-879). doi: 10.1145/1390156.1390266

Sangwan, N., & Bhatnagar, V. (2020). Comprehensive contemplation of probabilistic aspects in intelligent analytics. *International Journal of Service Science, Management, Engineering, and Technology (IJSSMET)*, *11*(1), 116-141. doi: 10.4018/IJSSMET.2020010108

Shivaprasad, D. L., Guru, D. S., Kavitha, R., & Saritha, N. (2022, December). Clustering based Indexing of Celebrity Cartoon Images for Retrieval. In *2022 Fourth International Conference on Cognitive Computing and Information Processing (CCIP)* (pp. 1-6). IEEE. doi:10.1109/CCIP57447.2022.10058621

Ye, F., Chen, C., & Zheng, Z. (2018, October). Deep autoencoder-like nonnegative matrix factorization for community detection. In *Proceedings of the 27th ACM international conference on information and knowledge management* (pp. 1393-1402). doi: 10.1145/3269206.3271697

Shaily Malik
Maharaja Surajmal Institute of Technology,
Delhi,
India
shaily.singh99@gmail.com
ORCID 0000-0003-1840-5412

Nishtha Jatana
Maharaja Surajmal Institute of Technology,
Delhi,
India
Nishtha.jatana@gmail.com
ORCID 0000-0002-6863-6756

Divya Jatain
Maharaja Surajmal Institute of Technology,
Delhi,
India
Divyajatain@msit.in
ORCID 0000-0002-4339-1828

Geetanjali Sharma
Maharaja Surajmal Institute of Technology,
Delhi,
India
gsharma@msit.in
ORCID 0000-0002-8906-7820

Vaani Garg
Maharaja Surajmal Institute of Technology,
Delhi,
India
vaani.garg@msit.in
ORCID 0009-0003-268-6755

Amita Yadav
Maharaja Surajmal Institute of Technology,
Delhi,
India
amitaay@gmail.com
ORCID 0000-0002-6680-7160