Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 218 (2023) 532–541

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Machine Learning and Data Engineering

A Deep Learning Approach for Classification of Medicare Beneficiaries Based on Gender and being Affected with Cancer

Manish K. Pandey[a,b], Karthikeyan Subbiah[b]

[a]*Centre for Quantitative Economics and Data Science, Birla Institute of Technology, Mesra, Ranchi-835215, Jharkhand, India*
[b]*Department of Computer Science, Institute of Science, Banaras Hindu University, Varanasi-221005, Uttar Pradesh, India*

Abstract

With the advent of the Third computing platform of Social Mobility Analytics and Cloud (SMAC), data is getting generated in huge amounts. This huge amount of data is collected for domain-specific information to process them to get required domain-specific information as in real-time health analytics, financial frauds, real-time automated car driving, vital information of patients undergoing robotic surgery, handling cyber threats etc. This huge data, also known as Big Data, is highly unstructured and imbalanced that is not possible for traditional techniques to handle and process. Advancements in computing power, speedy data storage and convergence of SMAC technologies have also contributed to the swift acceptance of the technology. This led to innovative analytical techniques that are data as well as computation intensive. One such technique is Deep Learning which originated from the artificial neural network and found its use in handling many real-life problems involving multidimensional features. The advantage of Feature Learning or Representational Learning makes Deep Learning a wonderful tool for big data analytics. The previous level of hierarchy transfers the feature learning to the next levels and thus complex features are learned through the learning of simpler features at different levels of abstraction. For efficient learning of these features, tuning of hyper-parameters is a mandatory step. The current work incorporates Grid Search for classification to find the best classifier for the classification of Medicare beneficiaries based on two scenarios. The first Scenario is beneficiaries who are affected by cancer and the Second Scenario is where Medicare beneficiaries are provided Gender wise (being a Female beneficiary). By experimenting using these algorithms at 10-fold cross-validation, the best results were achieved in the sensitivity of 99.17 %, Specificity of 97.68 % and accuracy of 98.8 % with Deep Learning Neural Network with Dropout for First Scenario and achieved the best results in the sensitivity of 82.97 %, Specificity of 68.71 % and accuracy of 75.05 % with Random Forest for Second Scenario.

Keywords: Deep Learning; DLNNWD; DLNNWOD; SMAC; Random Forest, SDG

1. Introduction

A lot has changed in computing from mainframe to client-server architecture to the SMAC era. SMAC stands for Social, Mobility, Analytics & Cloud. The union of SMAC technologies commences a new era known as the third computing paradigm [1,2] that gives rise to the gigantic growth of varied data. The data is generated from a diversified application extending from social sites, mobile apps, real-time streams, smart devices, RFID chips, Internet of Everything [IoE] etc [3], Healthcare Streams [1,4,5] etc. The enhancement in Information Security Management System [ISMS] standards [6] in managing the security challenges confronted by Cloud Service Providers [CSP] has

10.1016/j.procs.2023.01.035

enhanced the confidence of cloud service users again helping in the deployment of numerous applications on the cloud [7]. The improvements in web services recommendations have also amplified the deployment of various applications on the cloud [8–13]. The convergence has enabled everyone to connect to everything, known as the Internet of Everything [IoE]. The biggest trend as far as the generation of data in this era is concerned, is Big Data. This data is humongous and could not be analyzed using traditional legacy-based systems. Thus, this convergence is found to be leading to a paradigm shift where the focus is on data as well as on computational resources. Both of these in combination with augmented analytics play a crucial role in knowledge discovery [14–19].

The increase in the size is accompanied by the growth in the number of features that made the importance of deep learning quite significant. The concept of Deep learning is associated with the classical neural network [NN]. What makes Deep learning different from traditional NN is its capability of using hidden neurons and layers. This capability of Deep Learning permits an exhaustive exposure of unstructured data into layers which gives an output that acts as the projection of the input data with reduced dimensionality. This is further utilized in generating abstraction of the features at a higher level. The early discussions regarding biological neurons go back to the late fifties when [20] proposed perceptron for binary classification. The neurons stimulate other neurons that create a network of neural activities that helps in encoding, processing, and transmitting the information. The Perceptron also emulates the biochemical process that led them to transfer the learning from an input layer to an output layer directly with the help of an activation function. Further improvement was made by [21] when epochs were utilized to solve a complex problem in intermediate steps using more than one hidden layer. Delta rule Learning was utilized, and the weights of the neuron were tuned for better results. The delta rule was implemented through back propagation [22]. The utilization of more hidden layers for creating neural networks facilitates the generation of deep architecture that is known as Deep Neural Networks [DNN]. To avoid failure of the learning process, training of DNNs is a must, as any error can lead to the vanishing of the gradient through backpropagation. This can be solved using advanced variants of backpropagation [23] but ultimately results in slow learning. The unique feature of DNN lies in choosing a different approach to train the data separately for supervised and unsupervised learning techniques. In the case of unsupervised technique, one need not label the data for training the DNN while in supervised learning the weights are utilized for predicting the target value through minimization of the errors generated during the training. The learning in DNN is made through hierarchical representation [24,25] that has attracted researchers from various streams to build state-of-the-art solutions such as speech recognition [26,27], image processing [28,29], collaborative filtering [30], voice-enabled services [31,32] etc. The concepts of machine learning, overfitting, errors minimization, learning weights etc. were discussed in detail by [33–35].

The true behaviour of DNNs was not tested to its fullest due to technological limitations. With the convergence of SMAC, the benefits of DNNs are utilized in several real-world applications. One such application is achieving health coverage universally. This was identified as a challenge globally and included in the sustainable development goals of the United Nation. This is included under Goal 3 of Good Health and Well-Being. The major hurdles in achieving health coverage by 2030 come from the protection of financial risk. One of the countries that have worked towards achieving this goal is the United States which is providing Health coverage in form of Medicare and tracking the beneficiaries to further identify and protect any financial risk involved to the beneficiaries. The current work utilized the Medicare claims data of patients that suffered from chronic conditions to identify beneficiaries gender-wise and those that suffered from cancer. To classify the beneficiaries, DNN was utilized with various regularization methods. The regularization methods such as dropout [36] were used to avoid the over-fitting problem. The classifiers Random Forest [RF] and deep learning neural networks with dropout [DLNNWD] and without dropout [DLNNWOD] were used for classification purposes. Before classification, optimal parameters were identified using the grid search strategy.

2. Materials and Methods

2.1 Data Set

The data set was collected from [37] and constitutes the Medicare claims that characterize the beneficiaries in terms of a profile which is a mix of personal details such as name, age, gender, chronic situations etc. The initial dataset has many missing values for which null value imputation has been applied. The method which was selected was maximum likelihood where the null values initially were all removed followed by identification of the columnar distribution. Using Measure of central tendency, missing values were imputed using the sampling points through the

distribution. Further the prominent features were identified as described in section 2.2.

2.2 Selection of Input Features Vectors

The 15 features from the profile are depicted in Figure 1. The rest of the variables are associated with providing enrolment information for the Medicare program as well as averages of cost and utilization from the claimed data set. The averages are evaluated and presented separately based on enrolment with and less than 12 months.

Figure 1 Feature Description of the data sets.

2.3 Proposed Methodologies

The Experiments were conducted separately using three different deep learning algorithms for the classification of Medicare beneficiaries based on two scenarios. The first Scenario is beneficiaries who are affected by cancer and the Second Scenario is where the classification was made Gender wise (being a Female beneficiary). The proposed methodology is depicted in Figure 2

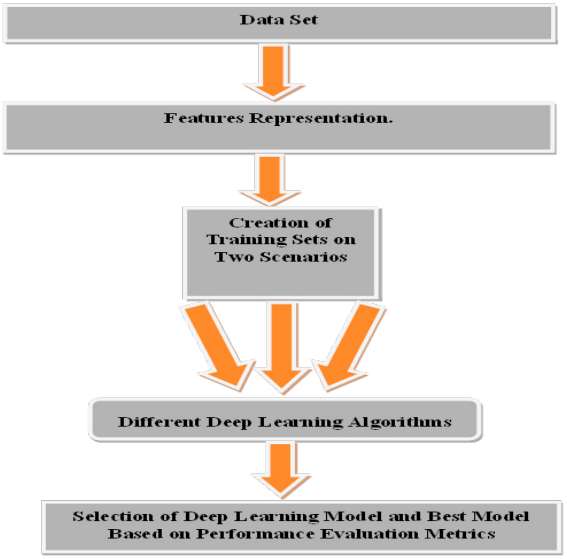

Figure 2 Flow diagram of the Proposed Methodology

2.4 Classification Protocol

The classifiers Random Forest and Deep Learning Neural Network with Dropout were found to be better performing for the proposed classification problem of Medicare beneficiaries based on being Cancer affected and gender-wise respectively.

2.5 Deep Learning Algorithms

Three deep learning algorithms were used for the classification of two Scenarios as mentioned in the proposed methodologies section.

2.5.1 Random Forest**:** An ensemble learning method that bases the learner's decision tree was utilized. In the first step, the bootstrapped instance was selected and trained using the base learner. In the next step, random instances were selected at every node for the evaluation. The algorithm stops once all the instances were trained using the base learner. The output is obtained as the fusion of all the individual base learners' outcomes. Three hyperparameters were used during experimentation that is, **sample rate** which constitutes the number of samples to be produced at every split, **ntrees** which is the number of trees and **max depth** which represents the depth, to which a tree is permitted to grow in turn, regulate the complexity of each tree.

2.5.2 Deep Learning Neural Networks:

More than one hidden layer in a neural network constitutes a Deep Learning Neural Network (DLNN) which is useful in building higher-level features out of lower-level features [38]. The network is prone to overfitting for which regularization technique is used. Overfitting refers to a situation in which the model statistically learns consistently from a specific training set. It means the models end up learning the inappropriate noise and perform inaccurately on the new dataset. Thus, recognizing overfitting would be achieved in the best manner by monitoring the loss and accuracy of both, the training set as well as the validation set. To encounter this situation reducing architectural complexity, batch normalization, regularization with dropout or weight decay [L2 regularization] and data augmentation are the preferred solutions. The current work incorporated regularization with the dropout technique for handling overfitting. Dropout works by selecting activations randomly and setting them to 0 during the training to reduce the sensitivity of the model to the specific weights in the network [39]. The weight decay imposes a penalty on the model's weight so that weights always take lesser values. To reduce the overfitting [40,41], the regularization term, also known as weight decay is added to the loss function E (θ). This is represented in the form

$$ER(\theta) = E(\theta) + \lambda\Omega(w) \quad (1)$$

Here, w is the weight vector, λ is the regularization factor (coefficient), and the regularization function Ω (w) is

$$\Omega(w) = \frac{1}{2}w^T w) \quad (2)$$

It is important to note that the biases are not regularized [42].

The advantage of DLNN is its ability to build higher-level features from lower-level features [38]. Deep learning neural networks with dropout (DLNNWD) and without dropout (DLNNWOD) were evaluated for the given dataset. To select the best DLNN model, the following hyperparameters are searched:

- **Input_dropout_ratios:** This quantifies the number of features accessible per training sample.
- **Hidden:** This controls the amount as well as the size of the hidden layer.
- **Regularization parameters:** As discussed earlier, these are used to normalize the overfitting problem. To regularize absolute weights and the sum of squared weight, L1 (Least Absolute Shrinkage Selection Operator) and L2 (Ridge Regression) are utilized respectively. L1 and L2 regularization on least-squares is represented in Equations 3 and 4 respectively.

$$w^* = \arg\min_w \sum_j \{(t(x_j) - \sum_i w_i h_i(x_j)\}^2 + \lambda \sum_{i=1}^{k} |w_i| \quad (3)$$

$$w^* = \arg\min_w \sum_j \{(t(x_j) - \sum_i w_i h_i(x_j)\}^2 + \lambda \sum_{i=1}^{k} w_{i^2} \quad (4)$$

- **hidden_dropout_ratios:** This parameter is responsible for controlling the quantity of the inputs that are available for the training of each hidden layer.
-
- **Activation functions or Transfer functions:** Popularly, Softmax and Rectifier, Maxout, and Tanh are used as activation functions for output and input hidden layers respectively. The training of the dataset is carried out for 10 epochs to avoid slow convergence. To compute the optimal values of the hyperparameters, the Grid Search technique was utilized with a random search strategy. This is implemented to ensure the completion of the grid search after 10 epochs. The hyperparameters and their corresponding values that were used in the current work are given in Figure 3.

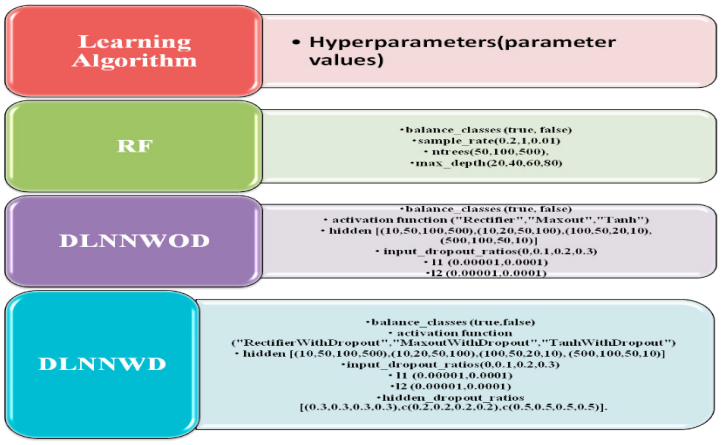

Figure 3 Values of the tuned Hyper-parameters

2.6 Performance Evaluation Metrics:

The relative performance of the classifiers was evaluated on a threshold basis. The number of correctly classified instances of being a cancer patient or female patient is termed as True Positive (TP). False Negative (FN)

is the incorrectly classified instance of being a cancer patient or female patient. The correctly classified instances of being a non-cancer patient or male patient are termed as True Negatives (TN) and the incorrectly classified instances of being a non-cancer patient or male patient are termed as False Positive (FP).

Sensitivity represents correctly predicted instances of being a cancer patient or female given by

$$Sensitivity = \frac{TP}{(TP+FN)} \times 100 \quad (5)$$

Specificity: This gives the ratio of correctly predicted Not Cancer/ Male instances and is given by

$$Specificity = \frac{TN}{(TN+FP)} \times 100 \quad (6)$$

Accuracy: The percentage of correctly predicted Cancer/Female and Not Cancer/ Male instances.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \times 100 \quad (7)$$

AUC represents the area under receiver operating characteristics (ROC) [43,44]. The classifier is predicting best if the value reaches 1.

MCC: A widely used parameter for performance evaluation, Mathew's correlation coefficient is described mathematically as

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \quad (8)$$

G-Means represents the geometric mean of sensitivity and specificity that signifies the balanced interpretation of accuracies and is described mathematically as

$$G - Means = \sqrt{Sensitivity * Specificity} \quad (9)$$

H2O package (45) is utilized for carrying out the experiments. The optimizer in H2O selects the threshold in such a manner that the best models would be chosen based on the F1 measure.

3. Result and Discussion

The experiment was conducted with three different deep learning algorithms, namely: 1) Deep Learning Neural Network with Dropout, 2) Deep Learning Neural Network without Dropout and 3) Random Forest on the Medicare beneficiary's data set, the performance observed through various metrics are listed in Table 1. 10-fold cross-validation as a hyperparameter was used to evaluate the generated models. Dropout is used to reduce the over-fitting of the neural network. Random Forest (RF) was found to be superior in classifying the beneficiaries as a cancer patient and DLNNWD was found to be superior in classifying the beneficiaries based on Gender. When deep learning algorithms were trained on data for the classification of Medicare beneficiaries, it resulted in the betterment of the performance using three algorithms. By experimenting using these algorithms at 10-fold cross-validation, the best results were achieved in the sensitivity of 99.17 %, Specificity of 97.68 % and accuracy of 98.8 % with DLNNWD for the First Scenario, i.e., classification based on being affected by Cancer and achieved the best results in the sensitivity of 82.97 %, Specificity of 68.71 % and accuracy of 75.05 % with RF for Second Scenario, i.e., classification based on Gender (being a Female beneficiary).

3.1 Observations on Hyperparameters: -

For hyperparameters in the case of RF, there is no impact on balance class and sample_rate in both scenarios. No trees are 500 in both the scenarios and max depth varied from 30 to 53 from First Scenario to Second Scenario. For hyperparameters in the case of DLNNWD, there is no impact on balance class, hidden_dropout_ratios, as well as input_dropout_ratios but hidden parameters with consecutive lower no of layers, are preferred in the case of the First Scenario. Also, l1 and l2 values are preferred with equal values. In the second scenario for DLNNWD, a balance class with false is to be preferred while hidden parameters with consecutive lower no of layers are preferred. Input_dropout_ratios in this scenario is 0 and hidden_dropout_ratios with value 0.3 is preferred. The lower value of l1 is preferred over l2. For Hyperparameters in the case of DLNNWOD, the True balance class and input_dropput_ratios of 0.1 is preferred in both Scenarios. The hidden parameter in increasing order of layers is preferred in First Scenario and the Hidden parameter in decreasing order of layers is preferred in the Second Scenario.

The value of l1 and l2 are preferred to be equal in the First scenario and the lower value of l1 is preferred over l2 in the Second Scenario. Activation_function of Maxout is preferred in the First scenario and Rectifier is preferred in the Second Scenario. A visual representation of all the classifiers in terms of receiver operating characteristics is given in Figures 4 to Figure 9.

Table 1 Performance Evaluation Metrics of Deep Learning Algorithms for classification of Medicare Beneficiaries based on being affected by Cancer and Gender

		Classification of Medicare Beneficiaries based on being affected by Cancer			Classification of Medicare Beneficiaries based on Gender		
Learning Algorithm	Hyper Parameters	The Best Values of Hyper Parameters	Performance Evaluation Metrics	10-Fold Cross-Validation	The Best Values of Hyper Parameters	Performance Evaluation Metrics	10-Fold Cross-Validation
RF	balance_class	True	Accuracy	98.55	TRUE	Accuracy	75.05
	max_depth	30	AUC	0.9975	30	AUC	0.837
	ntrees	500	Sensitivity	99.38	500	Sensitivity	82.97
	sample_rate	1	Specificity	96.1	1	Specificity	68.71
			G-Mean	97.72		G-Mean	75.5
DLNNWOD	balance_class	True	Accuracy	96.69	TRUE	Accuracy	51.38
	hidden	[10, 50, 100, 500]	AUC	0.9931	(100,50,20,10)	AUC	0.577
	input_dropout_ratios	0	Sensitivity	97.92	0.1	Sensitivity	93.34
	l1	0.00001	Specificity	93.04	0.0001	Specificity	18.34
	l2	0.00001	G-Mean	95.44	0.00001	G-Mean	41.37
	activation_function	Maxout			Rectifier		
DLNNWD	balance_class	True	Accuracy	98.8	FALSE	Accuracy	62.23
	hidden	[500,100,50,10]	AUC	0.996	(100,50,20,10)	AUC	0.712
	hidden_dropout_ratios	[0.3, 0.3, 0.3, 0.3]	Sensitivity	99.17	(0.3,0.3,0.3,0.3)	Sensitivity	87.47
	input_dropout_ratios	0.00	Specificity	97.68	0	Specificity	41.9
	l1	0.00001	G-Mean	98.42	0.0001	G-Mean	60.53
	l2	0.00001			0.00001		
	activation_function	MaxoutWithDropout			RectifierWithDropout		

Figure 4 AUC for the classification based on being affected from Cancer using DLNNWD

Figure 5 AUC for the classification based on being affected from Cancer using DLNNWOD

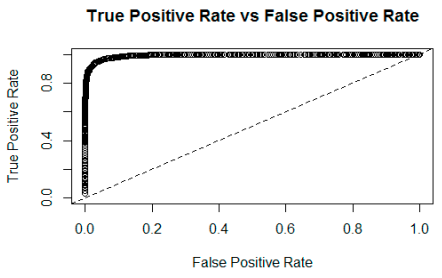

Figure 6 AUC for the classification based on being affected by Cancer using RF

Figure 7 AUC for the classification based on gender using DLNNWD

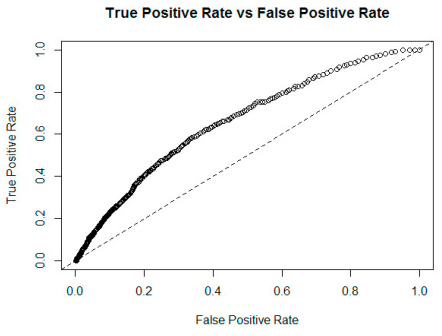

Figure 8 AUC for the classification based on gender using DLNNWOD

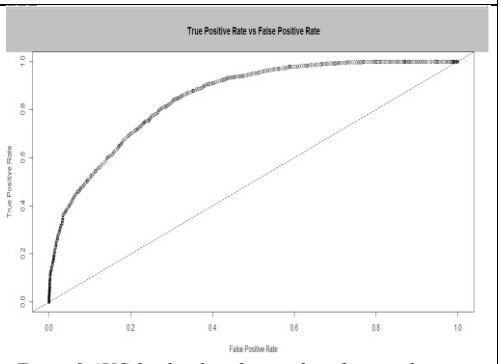

Figure 9 AUC for the classification based on gender using RF

4. Conclusion

The hierarchical learning process enables complex features to be learned and extracted as a data representation. This learning permits an exhaustive exposure of unstructured data into layers that gives an output which acts as the projection of the input data with reduced dimensionality. This is further utilized in generating abstraction of the features at a higher level. Due to the nature of huge data, tuning of Hyper-parameters is also mandatory for identifying and predicting the features. One such approach to tuning Hyper-parameters is grid search. In this paper, the Grid Search approach was applied for classifying Medicare beneficiaries based on Gender and being affected by Cancer. to find the best-performing algorithms Random Forest (RF), Deep Learning Neural Network without Dropout (DLNNWOD) and Deep Learning Neural Network with Dropout (DLNNWD) were used as the classifiers. The output

of the current experiments resulted in the efficient classification of Medicare beneficiaries. With this result, it is clear that accurate information of Medicare beneficiaries based on Age, gender, and disease, towards which the fund was spent, can provide an insight to the Government for efficient prediction of spending made towards specific ailment, age-wise or gender-wise or based on the person being affected with the chronic disease. This could help in the prevention of misuse of this fund by outlier detection as well as help the government to initiate programs to control the associated diseases by running preventive measures in early phases. The model could further be utilized to identify the claims that should be monitored on priority basis that could trigger some epidemic or the events that could be identified at a local level. This not only would help the authorities in offering informed decision making but also help in identification of several other parameters which need to be analyzed for improvisation of the models as well as the association of these parameters for understanding the root cause triggering the fraudulent claims. The limitation of the current work lies in the lack of appropriate mapping of the patients due to privacy concerns.

References

1. Pandey MK, Subbiah K. Performance analysis of time series forecasting using machine learning algorithms for prediction of ebola casualties. In: Communications in Computer and Information Science. Springer Verlag; 2018. p. 320–34.
2. Pandey MK. Novel Application Oriented Problem Solving Approaches in SMAC [Internet]. Banaras Hindu University; 2017. Available from: http://hdl.handle.net/10603/268444
3. Pandey MK, Subbiah K. Social networking and big data analytics assisted reliable recommendation system model for internet of vehicles. Vol. 10036 LNCS, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Springer Verlag; 2016. 149–163 p.
4. Pandey MK, Subbiah K. A Novel Storage Architecture for facilitating Efficient Analytics of Health Informatics Big Data in Cloud. Proc - 2016 16th IEEE Int Conf Comput Inf Technol CIT 2016, 2016 6th Int Symp Cloud Serv Comput IEEE SC2 2016 2016 Int Symp Secur Priv Soc Netwo. 2016;
5. MK Pandey SK. Performance analysis of time series forecasting of ebola casualties using machine learning algorithm. 2017;2:885–98.
6. Pandey MK, Kumar S, Karthikeyan S. Information Security Management System (ISMS) Standards in Cloud Computing-A Critical Review. 2013.
7. Kumar S, Pandey MK. Comparison of Dynamic Load Balancing Policies in Data Centers. Vol. 104, International Journal of Computer Applications. 2014.
8. Kumar S, Pandey MK, Nath A, Subbiah K, Singh MK. Comparative study on machine learning techniques in predicting the QoS-values for web-services recommendations. In: International Conference on Computing, Communication and Automation, ICCCA 2015. Institute of Electrical and Electronics Engineers Inc.; 2015. p. 161–7.
9. Kumar S, Pandey MK, Nath A, Subbiah K. Performance Analysis of Ensemble Supervised Machine Learning Algorithms for Missing Value Imputation. In: Proceedings - International Conference on Computational Intelligence and Networks. Institute of Electrical and Electronics Engineers Inc.; 2016. p. 160–5.
10. Kumar S, Pandey MK, Nath A, Subbiah K. Missing QoS-values predictions using neural networks for cloud computing environments. In: 2015 International Conference on Computing and Network Communications, CoCoNet 2015. Institute of Electrical and Electronics Engineers Inc.; 2016. p. 414–9.
11. Singh VP, Pandey MK, Singh PS, Karthikeyan S. An empirical mode decomposition (EMD) enabled long sort term memory (LSTM) based time series forecasting framework for web services recommendation. In: Frontiers in Artificial Intelligence and Applications. IOS Press; 2019. p. 715–23.
12. Singh VP, Pandey MK, Singh PS, Karthikeyan S. Neural Net Time Series Forecasting Framework for Time-Aware Web Services Recommendation. Procedia Comput Sci [Internet]. 2020;171:1313–22. Available from: https://linkinghub.elsevier.com/retrieve/pii/S1877050920311194
13. Singh VP, Pandey MK, Singh PS, Karthikeyan S. An Econometric Time Series Forecasting Framework for Web Services Recommendation. In: Procedia Computer Science [Internet]. Elsevier B.V.; 2020. p. 687–702. Available from: https://www.cys.cic.ipn.mx/ojs/index.php/CyS/article/view/3402
14. Lin J, Kolcz A. Large-scale machine learning at Twitter. Proc ACM SIGMOD Int Conf Manag Data. 2012;793–804.
15. Smola A, Narayanamurthy S. An architecture for parallel topic models. Proc VLDB Endow. 2010;3(1):703–10.
16. Chu CT, Kim SK, Lin YA, Yu YY, Bradski G, Ng AY, et al. Map-reduce for machine learning on multicore. Adv Neural Inf Process Syst. 2007;281–8.
17. Panda B, Herbach JS, Basu S, Bayardo RJ. Map reduce and its application to massively parallel learning of decision tree ensembles. Scaling up Mach Learn Parallel Distrib Approaches. 2011;9780521192:23–48.
18. Mittal, M., Sharma, R. K., & Singh VP. Modified single pass clustering with variable threshold approach. Int J Innov Comput Inf Control. 2015;11(1):375–86.
19. Pandey MK, Saini A, Subbiah K, Chintalapudi N, Battineni G. Improved Carpooling Experience through Improved GPS Trajectory Classification Using Machine Learning Algorithms. Information [Internet]. 2022 Aug 3;13(8):369. Available from: https://www.mdpi.com/2078-2489/13/8/369
20. Rosenblatt F. The Perceptron - A Perceiving and Recognizing Automaton. Report 85, Cornell Aeronautical Laboratory. 1957. p. 460–1.
21. David E. Rumelhart JLM and PRG. Parallel distributed processing. Cambridge, MA, USA: MIT Press; 1986.
22. D. E. Rumelhart, G. E. Hinton and RJW. Learning representations by back-propagating errors. In: Rosenfeld JAA and E, editor. Neurocomputing: Foundations of research. Cambridge, MA, US: MIT Press; 1988. p. 696–9.
23. Le Q V., Ngiam J, Coates A, Lahiri A, Prochnow B, Ng AY. On Optimization Methods for Deep Learning Quoc. In: Proceedings of the 28th International Con- ference on Machine Learning. Bellevue, WA, USA; 2011.

24. Bengio Y, Bengio S. Modeling high-dimensional discrete data with multi-layer neural networks. Adv Neural Inf Process Syst. 2000;400–6.
25. Ranzato M, Boureau YL, Le Cun Y. Sparse feature learning for deep belief networks. Adv Neural Inf Process Syst 20 - Proc 2007 Conf. 2009;(Mcmc):1–8.
26. Dahl GE, Yu D, Deng L, Acero A. Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. IEEE Trans Audio, Speech Lang Process. 2012;20(1):30–42.
27. Santana LMQ De, Santos RM, Matos LN, Macedo HT. Deep Neural Networks for Acoustic Modeling in the Presence of Noise. IEEE Lat Am Trans. 2018;16(3):918–25.
28. Ciresan DC, Meier U, Gambardella LM, Schmidhuber J. Deep, Big, Simple Neural Nets for Handwritten Digit Recognition. Neural Comput. 2010;22(12):3207–20.
29. Zeiler MD, Taylor GW, Fergus R, Tang S, Goto S, Hodgson AJ. Adaptive Deconvolutional Networks for Mid and High Level Feature Learning. Proc - IInternational Conf Comput Vis. 2011;2018–25.
30. Salakhutdinov R, Mnih A, Hinton G. Restricted Boltzmann Machines for Collaborative Filtering. In: Proceedings of the 24 th International Confer- ence on Machine Learning. Corvallis, OR; 2007. p. 245–54.
31. Amoore L. The Learning Machines: In: Cloud Ethics. 2020. p. 56–82.
32. Y. Wang, D. Yu, Y. Ju and AA. "Voice search" in Language Understanding. In: G. Tur and R. De Mori, editor. Language Understanding: Systems for Extracting Semantic Information From Speech. New York, NY, USA: Wiley; 2011. p. 480.
33. Pedro Domingos. A Few Useful Things to Know About Machine Learning. Commun ACM [Internet]. 2012;55(10):79–88. Available from: https://dl.acm.org/citation.cfm?id=2347755
34. Vapnik VN. An overview of statistical learning theory. IEEE Trans Neural Networks. 1999;10(5):988–99.
35. Bishop C. Pattern Recognition and Machine Learning. 1st ed. Springer-Verlag New York; 2006. XX, 738.
36. Nitish Srivastava GH, Alex Krizhevsky IS, Ruslan Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. J Mach Learn Res 15. 2014;15:1929–58.
37. CMS. Chronic Conditions Public Use File [Internet]. 2010 [cited 2020 Oct 16]. Available from: https://www.cms.gov/Research-Statistics-Data-and-Systems/Downloadable-Public-Use-Files/BSAPUFS/Chronic_Conditions_PUF.html
38. Lecun Y, Bengio Y, Hinton G. Deep learning. Nature. 2015;521(7553):436–44.
39. Hinton GE, Srivastava N, Krizhevsky A, Sutskever I, Salakhutdinov RR. Improving neural networks by preventing co-adaptation of feature detectors. 2012;1–18. Available from: http://arxiv.org/abs/1207.0580
40. Kingma DP, Ba JL. Adam: A method for stochastic optimization. 3rd Int Conf Learn Represent ICLR 2015 - Conf Track Proc. 2015;1–15.
41. Pascanu R, Mikolov T, Bengio Y. On the difficulty of training recurrent neural networks. 30th Int Conf Mach Learn ICML 2013. 2013;(PART 3):2347–55.
42. Glorot X, Bengio Y. Understanding the difficulty of training deep feedforward neural networks. J Mach Learn Res. 2010;9:249–56.
43. Ling CX, Huang J, Zhang H. AUC: A better measure than accuracy in comparing learning algorithms. Lect Notes Comput Sci (including Subser Lect Notes Artif Intell Lect Notes Bioinformatics). 2003;2671:329–41.
44. Huang J, Ling CX. Using AUC and accuracy in evaluating learning algorithms. IEEE Trans Knowl Data Eng. 2005;17(3):299–310.
45. H2O [Internet]. Available from: http://h2o.ai