OPTIMIZED DEEP LEARNING CNN MODEL USING PARAMETRIC OPTIMIZATION FOR BIOMEDICAL ENTITY IDENTIFICATION FROM PUBMED ARTICLES

G.SUGANYA
Ph.D Research Scholar
Department of Computer Science
Bharathiar University
Coimbatore – 641046
suganya.cs@buc.edu.in

R.PORKODI
Associate Professor
Department of Computer Science
Bharathiar University
Coimbatore – 641046
porkodi_r76@buc.edu.in

***Abstract*— Deep learning is a subset of machine learning and mostly used algorithms to find out complex relationships between various entities appeared in the dataset. Simple Deep Learning (DL) model contains 3 major layers such as input, hidden and output layer. Biomedical Entities (gene, mutation and drug) are extracted by using Deep Learning parametric optimization technique with input size (convolutional layer), filter kernel size, number of neurons, optimizer, learning rate and layers to develop optimal CNN model. In which two CNN models are developed and compared with themselves. The study experimented with Alzheimer disease which contains of 100 full-text articles are collected from PubMed database. The result shows that the developed CNN model obtained 81%, 81%, 86% and error rate of 1.2, 1.4, 1.2 for gene, mutation, drug names respectively. From this analysis, CNN M2 model obtained the better and good biomedical entities identification accuracy with convolutional layer size as 32*32, max pooling layer, 0.5 as learning rate, adam as optimizer with 500 epochs.**

Keywords — *Deep Learning, Convolutional Neural Network, Alzheimer disease, PubMed, Biomedical entities.*

I. INTRODUCTION

Deep learning (DL) refers to a class of machine learning techniques that exploit many layers of non-linear information processing for supervised or unsupervised feature extraction and transformation and for pattern analysis and classification. Deep learning networks can be categorized into i) Deep Neural Networks (DNNs), ii) Convolutional Neural Networks (CNNs)

iii) Recurrent Neural Networks (RNNs), iv) Restricted Boltzmann Machines (RBMs), v) Deep Belief Networks (DBNs) model (Liu, Chen, Jagannatha, & Yu, n.d.).

Deep Learning retrieved a lot of information in last years for its capacity to generalize models without the need of features and its ability to provide performance. Good performance can be achieved by accurately designing the architecture used to performance the learning tasks. To construct a comprehensive system to process the system, need of different modules like word-level modules which includes Part-Of-Speech (POS) and NER, sentence-level modules like dependency parsing and role labeling, and document-level modules like classification, and summarization.

Recently, Deep learning models have been applied to many natural language processing (NLP) tasks such as POS tagging, named entity recognition, semantic role labeling and sentiment analysis. For natural language processing, to achieve a precise and in-depth semantic understanding, it is impossible to solve the essential problems simply by relying on data annotation and computing power input. The massive text data information contains lot of valuable knowledge, to obtain the useful information, involve key research problem of artificial intelligence- knowledge acquisition (Zhang, Dai, & Jiang, 2020).

II. EXISTING SYSTEM

(Wagh, Kulkarni, Pawar, Kirange, & Kashid, 2017) proposed a Biomedical namedentity Recognition (BM-NER) system for extracting Name, Problem and Test from the Textual Clinical Lab reports. (Malarkodi, Lex, & Devi, 2016) presented the Named Entity Recognition System based on CRF (Conditional Random Field) that used to extract the entities like crop names, fertilizers, climate, location in the agricultural domain. (Settles, 2005) proposed a tool called ABNER (A Biomedical Named Entity Recognizer) and it is an open source software and user-friendly interface. The method was based on the CRF (Conditional Random Field) algorithm and it tested with two corpus such as NLPBA and BioCreative. (Yu, 2011) proposed various type of Long short term memory based models for sequence tagging. The model includes LSTM networks, bidirectional LSTM (Bi-LSTM), LSTM with Conditional Random Forest (CRF) layer (LSTM- CRF) and bidirectional LSTM with CRF layer (Bi-LSTM-CRF). (Sharma, 2020) proposed deep neural network for named entity recognition for Hindi language which based on convolutional neural network (CNN), bidirectional long short term memory (Bi-LSTM) and Conditional Random Field (CRF). (Cho & Lee, 2019) developed NER system for biomedical entities by incorporating n-grams with bi-directional long short-term memory (Bi-LSTM) and CRF which is contextual long short-term memory network (CLSTM). (Wang, Wang, Zhang, & Yan, 2016) proposed a framework based on convolutional neural networks (KPCNN) which combines explicit and implicit representations for short text classification. (Kim, 2014) proposed convolutional neural networks for sentence level classification tasks which trained on pre-trained word vectors. (Kumar, Kumar, & Soman, 2018) developed deep

learning model based on POS tagging for Malayalam Tweets. The tagged data was evaluated using deep learning methods like Recurrent Neural Network (RNN), gated Recurrent Units (GRU), Long Short-Term Memory (LSTM) and Bidirectional LSTM (Bi-LSTM). (Semberecki, 2017) developed a method of classification of articles using deep neural network with Long Short- Term Memory (LSTM) units.

III. CNN MODEL BY USING PARAMETRIC OPTIMIZATION

The proposed models are created from analyzing of hyper parameters in Deep Learning based Convolutional Neural Network. The model adopts themselves to learn the hierarchy of deep convolutional features and classify from the training data. The proposed CNN model process are divided into 2 modules namely feature extraction and prediction of entities as shown in Figure 1.

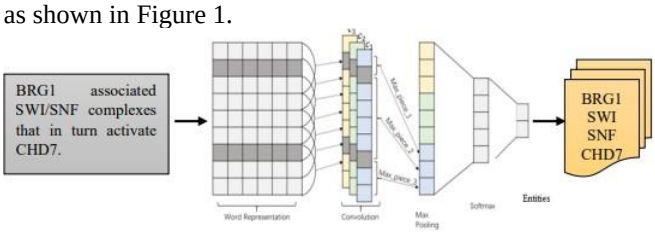


Fig. 1. Convolutional Neural Network Structure

CNN training algorithm mainly includes the following four steps:

Step 1 *Select a sample (X, Y) from sample set and enter X into network.*

Step 2 *After operation of network model, output actual result O.*

Step 3 *Calculate the difference between actual output O and ideal output Y.*

Step 4 *Backward propagation according to the minimization error, adjust corresponding weight matrix. Among them, the steps 1 and 2 are forward transmission phases, while steps 3 and 4 are reverse transmission phases (feedback phase).*

Table 1. CNN Hyper parameter

Layer Details		Description
Number of Layers	M1: 16*16	M2: 32*32
Pooling Layer	Max Pooling (1*1)	
Sequence length	1000	
Hidden layer	128	
Batch Size	128	
Number of iterations or Epochs	500	
Learning rate	0.5	
Optimizer	Adam()	

The CNN model contained of three layers in which one for convolutional layer and two for fully connected layer. Table 1 shows that the parameter details of the proposed CNN model. The fully connected layer has been fed to softmax layer which includes 3 categories which generates the distribution of individual categories. In CNN, there are 2 different models are created based on the different parameter. All the models are created with Max Pooling layer, Adam() as optimizer, learning rate as 0.5 and number of iterations as 500; the model differentiated with convolutional layer size.

IV. RESULTS AND DISCUSSION

A. *DATASET DESCRIPTION*

The input dataset for the proposed work consists of PubMed full-text articles related to Alzheimer disease in Homosapiens, which are downloaded from NCBI database. Data is based on the 3 categories namely Gene, Drug and Mutation. Totally 100 articles are used for all classes in which 70 and 30 articles are used for training and testing process.

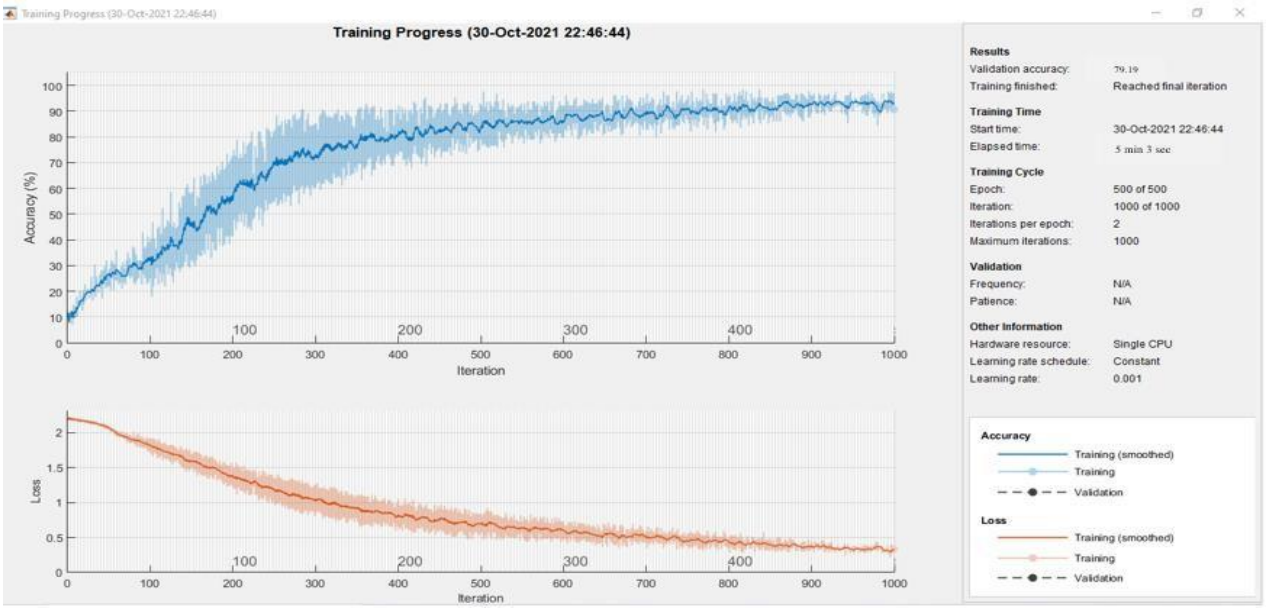
B. *PARAMETRIC OPTIMIZATION RESULTS BETWEEN MODELS FOR CNN*

The developed models are evaluated by prediction accuracy and loss rate for given full-text articles which is used to find best model with parameter optimization. From this evaluation Model M2 has performed better when compared with other models developed M1 model obtained the entities prediction accuracy 79.19% as Gene, 76.96% as Drug and 79.79% as Mutation. Likewise, M2 model obtained 80.62% as Gene, 86.21% as Drug and 81.62%as Mutation respectively. The Table 2 shows the obtained prediction accuracy, error rate and execution time of different models.

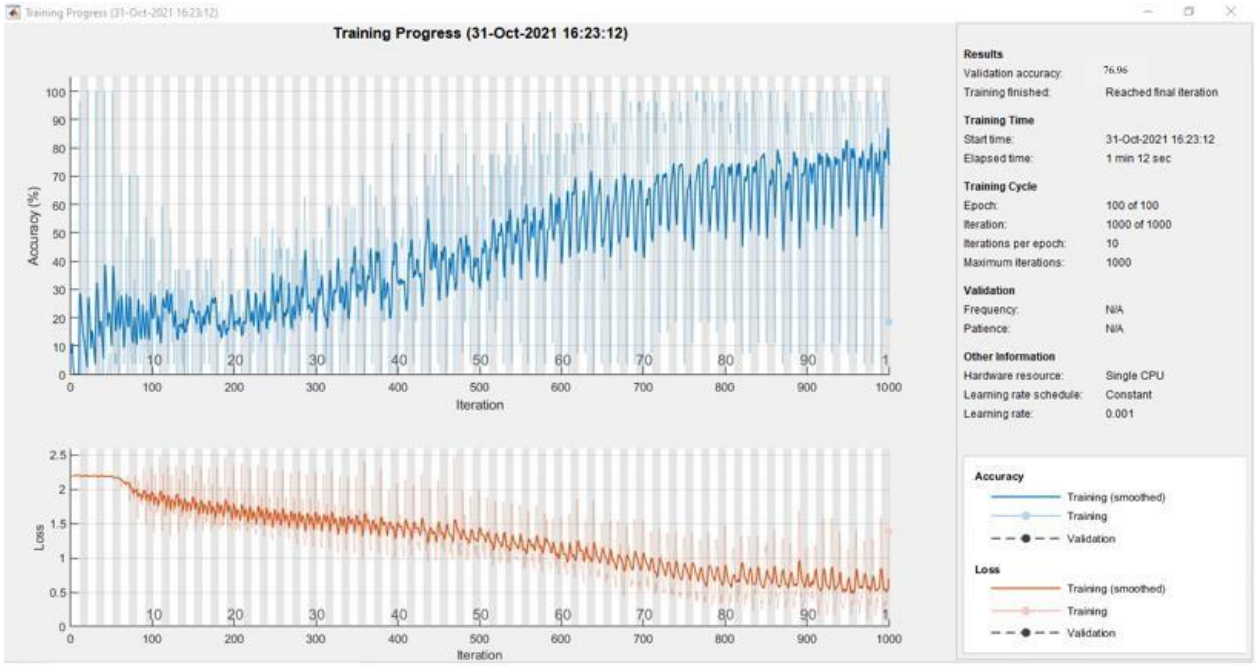
Table 2. CNN 2 models with Performance measures for Biomedical entities identification

Model Name	Metric	Gene (%)	Drug (%)	Mutation (%)
M1	Accuracy (%)	79.19	76.96	79.79
	Loss Rate	1.5	1.5	1.5
	Execution time	5 min 3 sec	1 min 12 sec	1 min 10 sec
M2	Accuracy (%)	80.62	86.21	81.62
	Loss Rate	1.2	1.2	1.4
	Execution time	3 min 23 sec	1 min 57 sec	1 min 18 sec

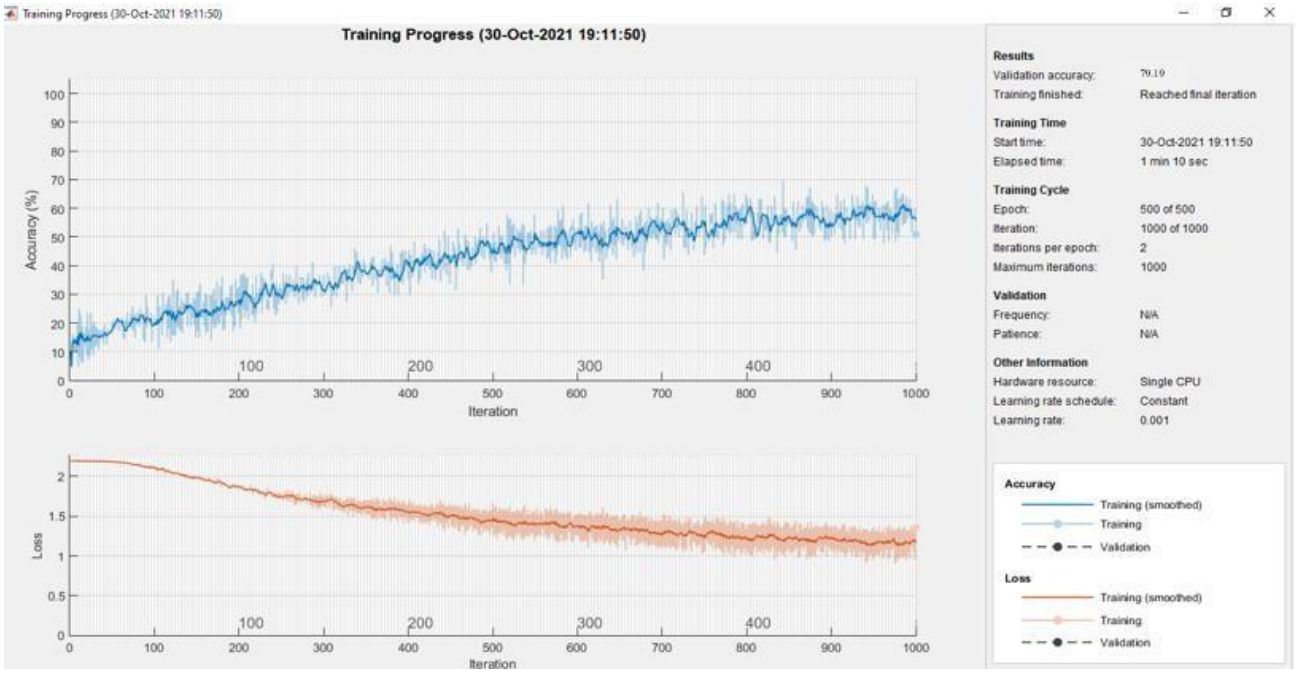
The performance of the proposed CNN model along with pre-trained networks for different set of training and testing for biomedical entities are shown in figure 2, 3 for M1, M2 model respectively. The proposed CNN models were compared with other models and obtained better result for those biomedical entities prediction.


Training Progress (30-Oct-2021 22:46:44)
Accuracy (%)
Iteration
Loss
Iteration
Results
Validation accuracy: 79.19
Training finished: Reached final iteration
Training Time
Start time: 30-Oct-2021 22:46:44
Elapsed time: 5 min 3 sec
Training Cycle
Epoch: 500 of 500
Iteration: 1000 of 1000
Iterations per epoch: 2
Maximum iterations: 1000
Validation
Frequency: N/A
Patience: N/A
Other Information
Hardware resource: Single CPU
Learning rate schedule: Constant
Learning rate: 0.001
Accuracy
Training (smoothed)
Training
Validation
Loss
Training (smoothed)
Training
Validation

a) Gene

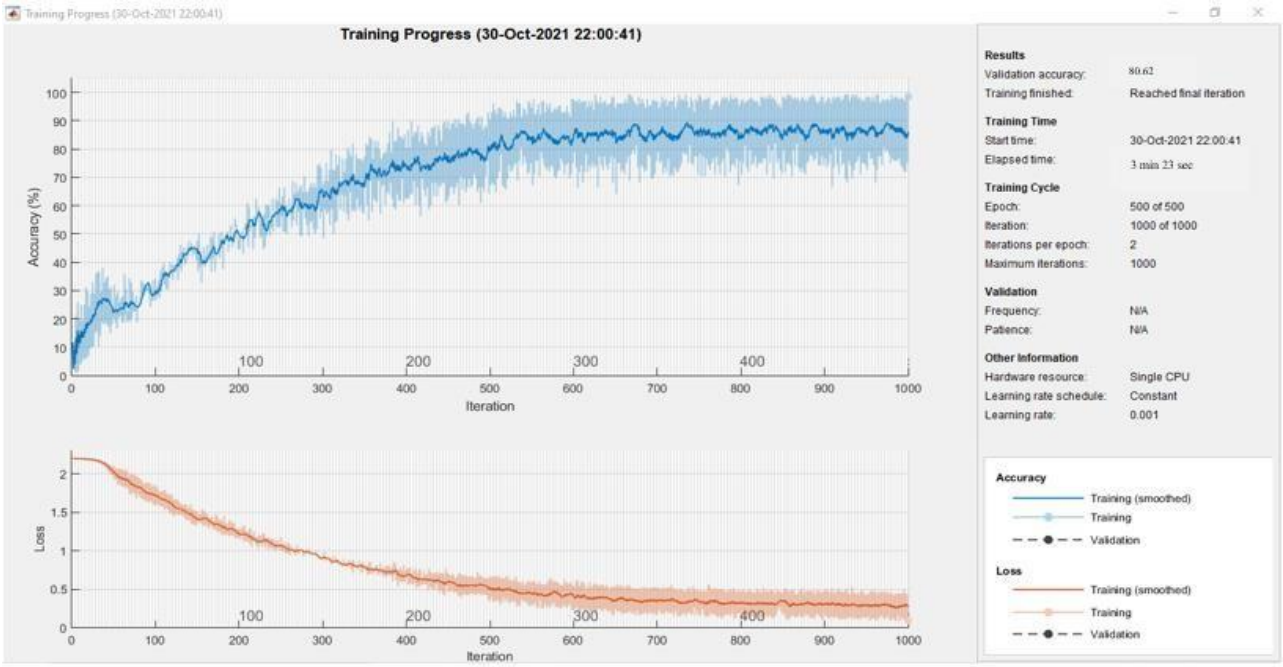

Training Progress (31-Oct-2021 16:23:12)
Accuracy (%)
Iteration
Loss
Iteration
Results
Validation accuracy: 76.98
Training finished: Reached final iteration
Training Time
Start time: 31-Oct-2021 16:23:12
Elapsed time: 1 min 12 sec
Training Cycle
Epoch: 100 of 100
Iteration: 1000 of 1000
Iterations per epoch: 10
Maximum iterations: 1000
Validation
Frequency: N/A
Patience: N/A
Other Information
Hardware resource: Single CPU
Learning rate schedule: Constant
Learning rate: 0.001
Accuracy
Training (smoothed)
Training
Validation
Loss
Training (smoothed)
Training
Validation

b) Drug

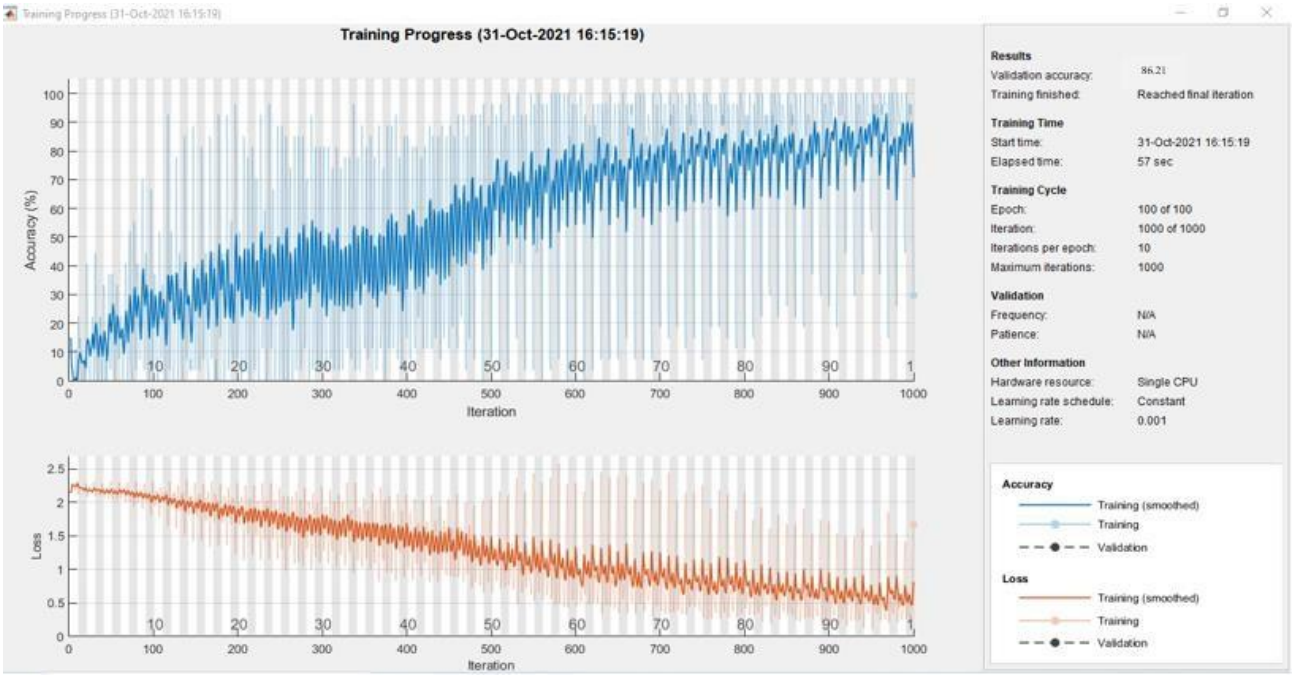


c) Mutation

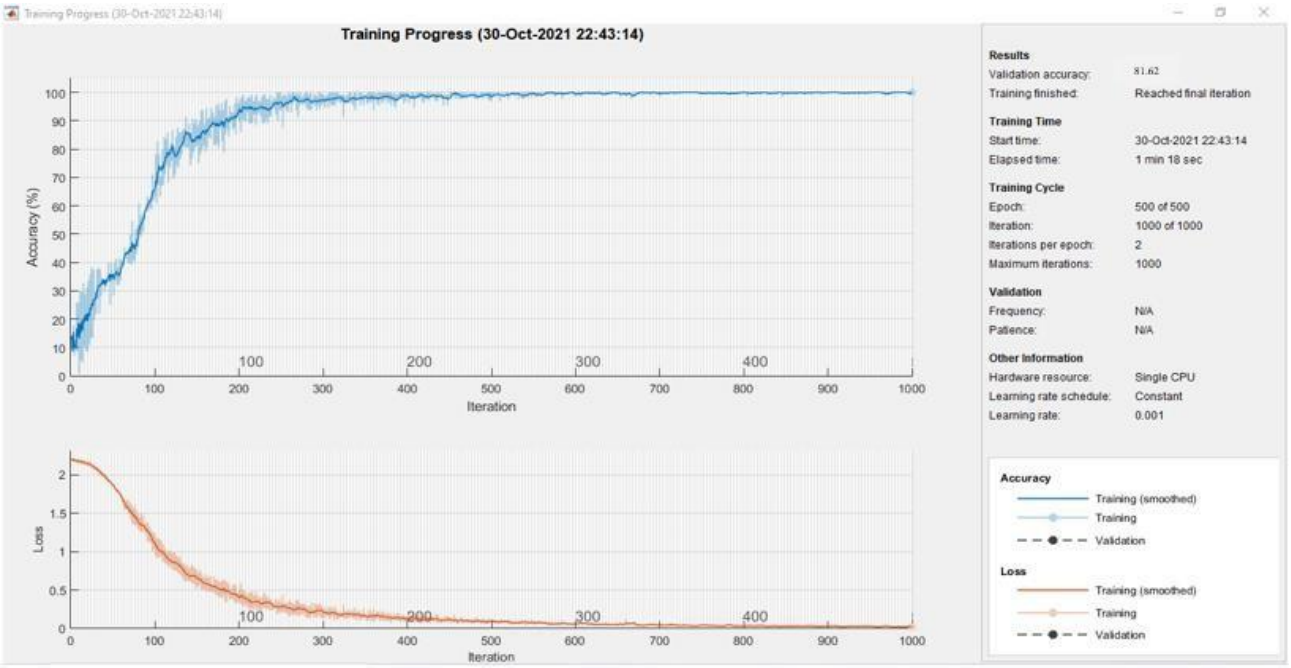
Fig. 2. M1 Model



a) Gene



b) Drug



c) Mutation

Fig. 3. M2 Model

V. CONCLUSION

In recent scenario, Deep Learning plays a vital role in the Biomedical entity identification. The study examined and analyzed the full-text PubMed articles to identify biomedical entities such as gene, drug and mutation. The proposed model is analyzing the impact of changing parameters with different number of layer sets. The input parameters considered for the optimization are number of layers, kernels, optimizer and activation function. These parameters are systematically validated to understand prediction accuracy. Finally, optimal values for these parameters for CNN model was identified through experimental study. This experimental result shows that the developed CNN model obtained prediction accuracy of 81%, 81%, 86% and error rate of 1.2, 1.4, 1.2 gene, mutation, drug names respectively. From this analysis, CNN M2 model obtained the better and good biomedical entities prediction accuracy.

References

[1] Liu, F., Chen, J., Jagannatha, A., & Yu, H. (n.d.). Learning for Biomedical Information Extraction : Methodological Review of Recent Advances Learning for Biomedical Information Extraction.

[2] Zhang, X., Dai, Y., & Jiang, T. (2020). A Survey Deep Learning Based Relation Extraction. https://doi.org/10.1088/1742-6596/1601/3/032029

[3] Wagh, K. S., Kulkarni, Aishwarya., Pawar, Pratiksha., Kirange, Neha., & Kashid, Shraddha. (2017). Bio Medical Named Entity Recognition Using Machine Learning Algorithms. International Journal of Advance Engineering and Research Development 1– 6.

[4] Malarkodi, C. S., Lex, Elisabeth., & Devi, Sobha. Lalitha Devi. (2016). Named Entity Recognition for the Agricultural Domain. Research in Computer Science, 117, 121–132.

[5] Settles, Burr. (2005). ABNER: An open source tool for automatically tagging genes, proteins and other entity names in text. Bioinformatics, 21(14), 3191–3192. https://doi.org/10.1093/bioinformatics/bti475

[6] Yu, K. (2011). Bidirectional LSTM-CRF Models for Sequence Tagging.

[7] Sharma, R. (2020). A deep neural network-based model for named entity recognition for Hindi language. Neural Computing and Applications, 32(20), 16191–16203. https://doi.org/10.1007/s00521-020-04881-z

[8] Cho, H., & Lee, H. (2019). Biomedical named entity recognition using deep neural networks with contextual information. 1–11.

[9] Wang, J., Wang, Z., Zhang, D., & Yan, J. (2016). Combining Knowledge with Deep Convolutional Neural Networks for Short Text Classification. 2915–2921.

[10] Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification. 1746–1751.

[11] Kumar, S., Kumar, M. A., & Soman, K. P. (2018). Deep Learning Based Part-of-Speech Tagging for Malayalam Twitter Data (Special Issue : Deep Learning Techniques for Natural Language Processing).

[12] Semberecki, P. (2017). Deep Learning methods for Subject Text Classification of Articles. 11, 357–360. https://doi.org/10.15439/2017F414