# 7. RECURRENT NEURAL NETWORK MODELS TO PREDICT ASD GENES, THEIR SUSCEPTIBILITY AND MUTATIONS

The Deep neural network implemented on the three different datasets presented in the previous chapter is good at learning abstract and hidden features from training data but is unable to remember past inputs as there is no memory associated with the model. To solve this issue and produce improved results of prediction, a kind of deep learning architecture, RNN is adopted at the next level. RNNs are connectionist models that explore the dynamics of sequences via cycles in the network of nodes. It holds a memory of previous inputs that persist in the internal state and influences the network output. RNNs retain a state that can represent information from an arbitrarily long context window and hence are suited well for sequence data analysis. RNN models are good at classifying variable length sequences and hence this work empirically evaluates the ability of RNN in the prediction task. This chapter elaborates the development of Bidirectional RNN (BRNN) based models to recognize ASD causing genes, their predisposition to ASD and the triggering mutations. The variants of RNN, such as LSTM and GRU are also experimented for the tasks mentioned above and demonstrated in this chapter.

## 7.1 BRNN BASED MODEL TO PREDICT ASD GENES, THEIR SUSCEPTIBILITY AND MUTATIONS

Standard RNNs process sequences in temporal order and ignore the future context. This problem is solved by Bidirectional recurrent neural networks [BRNN] by Schuster and Paliwal [98]. In the case of problems dealing with sequences it is beneficial to have access to future as well as past context. The basic idea of BRNNs is to present each training sequence forwards and backwards to two separate recurrent hidden layers, both of which are connected to the same output layer. This structure provides the output layer with complete past and future context for every point in the input sequence, without displacing the inputs from the relevant targets. Hence Bidirectional RNNs are found to be more suitable to solve the problem of identifying ASD causing genes, determining their susceptibility to ASD and recognizing the triggering mutations. This work deals with the development of three autonomous BRNN models trained using CMDS, MDs and GSDS datasets for these tasks.

**Methodology**

The building blocks of the proposed BRNN models include i) input logic ii) process logic iii) output logic and the architecture of this model is depicted in Fig.7.1. The input logic deals with the simulation of mutated gene sequences, preparation of the corpus and development of the dataset. The process logic involves the building of models for the respective tasks. The output logic deals with testing and performance evaluation of the models for their predictive accuracy.

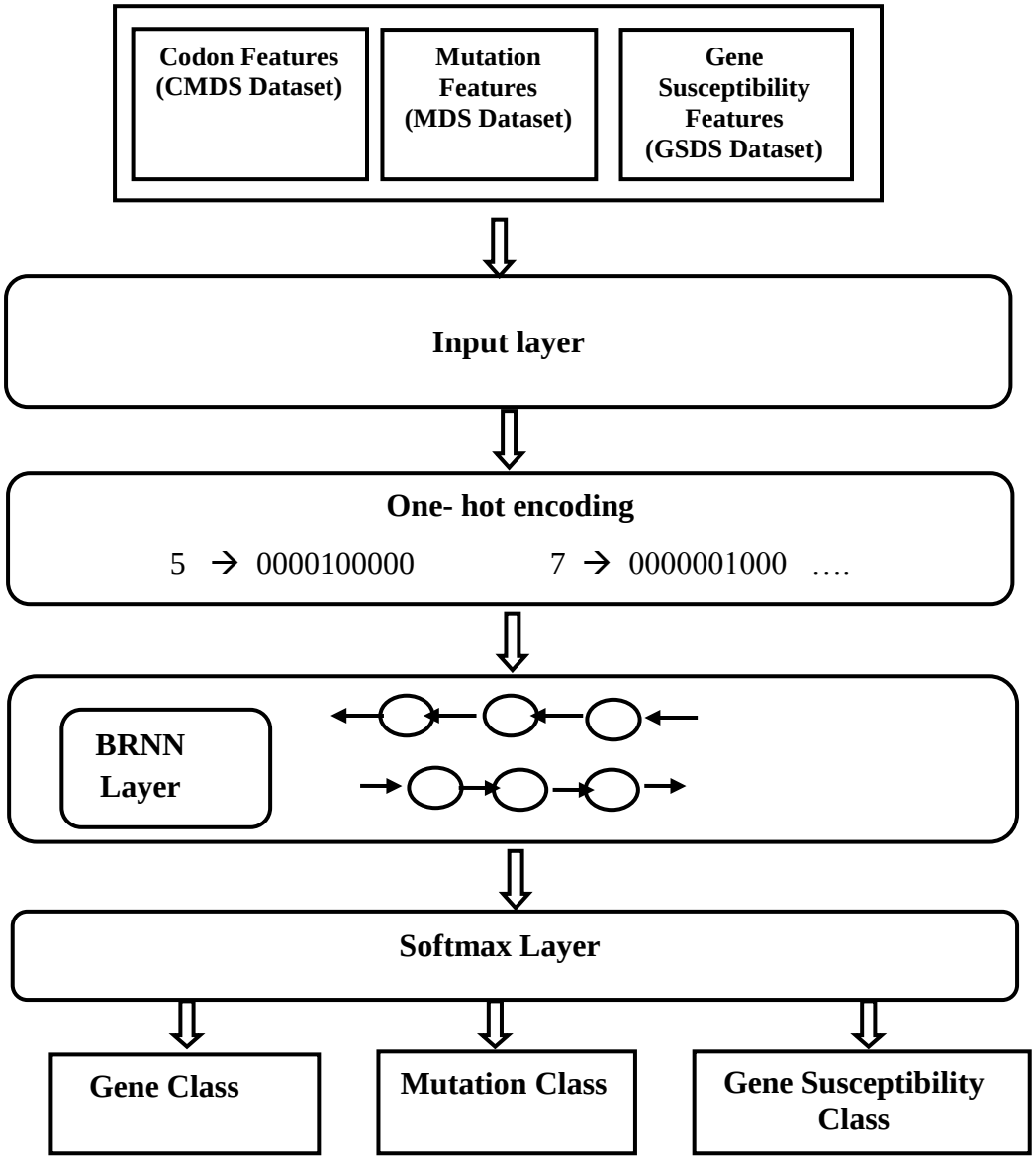


**Fig. 7.1 Architecture of the BRNN Based Prediction Model**

In the first phase, the corpus developed using 1000 mutated gene sequences accounting for ten types of ASD genes and four types of mutations described in Chapter 3 is used. The three datasets CMDS, MDS and GSDS described in Chapter 4 and 5 are used to build the models for classification of genes, mutations and susceptibility prediction. One hot encoding of the class values is used where each value is represented by a binary vector. For example class 5 is converted into 0000100000 and 6 is converted into 0000010000. The feature vectors are reshaped into a format that can be used as input to the BRNN.

The process component involves the construction of BRNN based classifiers. In this architecture, there is one input layer, 2 hidden layers with 8 memory units and an output layer.. The two hidden layers are distinguished by the recurrent connections from the past time steps and in the second layer the direction of recurrent connections is flipped, where activation is sent backwards along the sequence. When an input sequence and a target sequence are provided, the BRNN is taught by back propagation after unfolding across time. During BRNN training backpropagation is done not only vertically like in a standard DNN but also horizontally over T-1 context layers, the activation at the current timestep is done based on the past activations of the hidden layers from T-1 timesteps. BRNN split the neurons of a regular RNN into two directions, one for the forward states and another for the backward states 99]. By employing two time directions simultaneously, input data from the past and future of the current time frame can be used to calculate the same output which is lacking in DNN. The BRNN model is able to learn relations between consecutive signals and identify any type of regularity in the input.

In the ASD gene classification problem, the model consists of 10 neurons in the output layer and in the mutation classification model there are 4 neurons. The output layer consists of 3 neurons for predicting gene susceptibility to ASD. Multiple hyperparameters such as batch size, epochs, learning rate and dropout are fine tuned to set the optimal configuration of the network. Three independent models for gene identification, mutation recognition and gene susceptibility prediction have been built by learning the Bidirectional Recurrent neural network with three datasets.

In the concluding phase, 10 - fold cross-validation technique is applied for testing and the predictive performance of the three models are evaluated using various metrics such as precision, recall, F - measure, accuracy, log loss, specificity.

**Experiment and Results**

The Bidirectional Recurrent neural network is implemented using Keras in Python with the three datasets as input to build the BRNN based prediction models. A batch size of 64 is used to segment the data during training and testing. The network is trained using varying epochs of 50,100,150, 200 and 250 and dropout values of 0.2 to 0.5 are considered. To incorporate the future and past information, RNN with bidirectional approach is used here. The model uses a learning rate of 0.01, the Adam Optimizer, 2 hidden layers of size 8, a dropout probability of 0.3 and batch size of 64. The RNN is bidirectional and hidden layers use the rectifier activation function which is applied to the weighted sum to compute the output values of the layer. Softmax activation function is employed in the output layer. The model is trained for 250 epochs on the training examples. With these parameter settings, the three classifiers have been built. The recognition rates of these classifiers are evaluated using 10- fold cross-validation technique. The performance of the models is evaluated on metrics like precision, recall, F-measure, accuracy and log loss to obtain insights about the suitability of these approaches for the objectives. The prediction results of the gene prediction model, mutation prediction model and gene susceptibility prediction model are tabulated. The epochwise accuracy of these three models is shown in Table XXII.

**Table XXII Epochwise Accuracy of BRNN Models**

| Epochs | Gene Prediction Model | Mutation Prediction Model | Gene Susceptibility Prediction Model |
|---|---|---|---|
| 50 | 75.3% | 76.1% | 73.9% |
| 100 | 78.2% | 75.8% | 74.1% |
| 150 | 79.6% | 76.2% | 76.4% |
| 200 | 80.5% | 78.0% | 78.2% |
| 250 | 81.3% | 78.5% | 80.9% |

From the results in Table XXII, it is seen that BRNN offers a striking performance improvement for gene classification when compared to the other tasks. There is more than 3% improvement in the ASD gene recognition task rather than the mutation classification task. The model performs equally well in identifying the ASD genes and classifying their susceptibility as

their accuracy values are 81.3% and 80.9% respectively. The epochwise log loss of the three models is shown in Table XXIII.

**Table XXIII Epochwise Log Loss of BRNN for Three Models**

| Epochs | Gene Prediction Model | Mutation Prediction Model | Gene Susceptibility Prediction Model |
|---|---|---|---|
| 50 | 0.9831 | 1.5194 | 1.8952 |
| 100 | 0.9912 | 1.6844 | 1.7816 |
| 150 | 0.8652 | 1.2719 | 0.9830 |
| 200 | 0.8817 | 1.1165 | 0.9488 |
| 250 | 0.8010 | 0.9615 | 0.9459 |

It can be seen from the above table that the log loss associated with classifying the ASD genes is 0.810 which is comparatively less when compared to that of 0.9615 for mutation prediction and 0.9459 for ASD gene susceptibility recognition. The model has achieved better performance by reducing its misclassifications of genes. It is also noticed that the log loss reduces with respect to increase in the epochs. The results of the various other performance metrics is shown in Table XXIV.

**Table XXIV Performance Results of BRNN Models**

| Epochs | Gene Prediction Model | Mutation Prediction Model | Gene Susceptibility Prediction Model |
|---|---|---|---|
| Precision | 0.80 | 0.78 | 0.80 |
| Recall | 0.77 | 0.82 | 0.81 |
| F- Measure | 0.78 | 0.80 | 0.80 |
| Accuracy | 81.3% | 78.5% | 80.9% |
| Correctly classified instances | 407 | 392 | 405 |
| Incorrectly classified instances | 93 | 108 | 95 |
| Specificity | 76.6% | 75.4% | 73.5% |

The result analysis indicates that BRNN based gene prediction model shows outstanding performance in identifying the causative ASD genes with precision, recall, F-measure and

accuracy values of 0.80, 0.77, 0.78 and 81.3% respectively. The BRNN model is also effective in predicting the susceptibility of ASD causing genes with a precision of 0.80 and recall of 0.81. The classifier performs fairly well in recognizing the triggering mutations with precision of 0.78 and recall of 0.82. The gene prediction model has appropriately identified 407 instances. The specificity of BRNN reaches an elevated score value of 76.6% for gene prediction model whereas it is 75.4% and 73.5 % for mutation prediction and susceptibility prediction models. The experimental results with respect to various parameters are illustrated in Fig.7.2 to Fig.7.7.

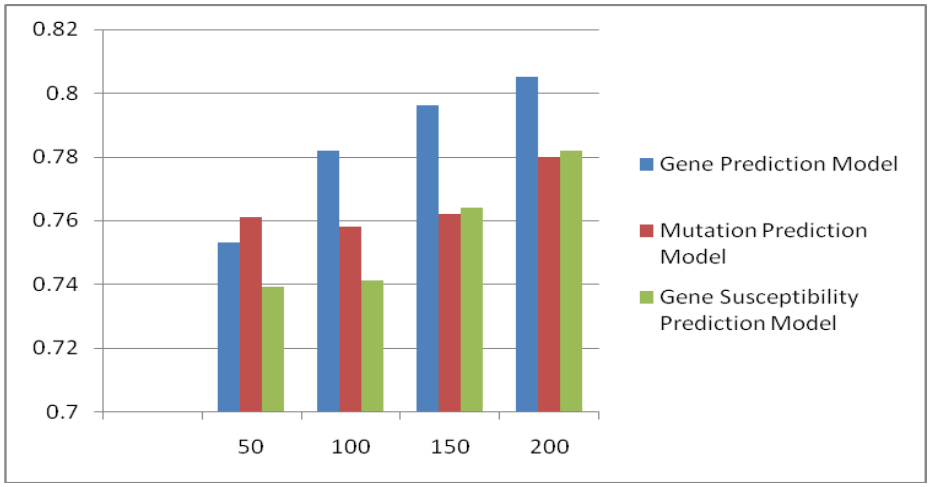


**Fig. 7.2 Accuracy of BRNN Models**

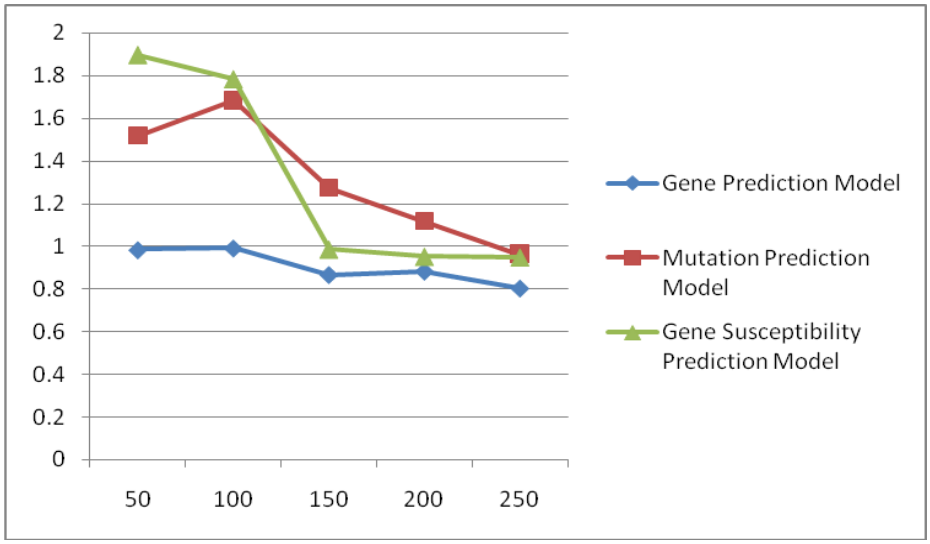


**Fig. 7.3 Epochwise Log Loss of BRNN Models**

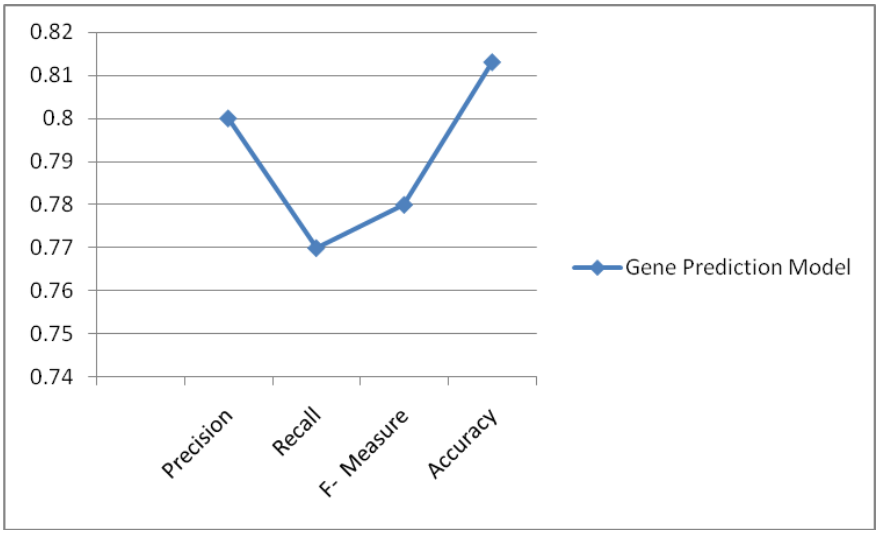


**Fig. 7.4 Performance Results of BRNN Based Gene Prediction Model**

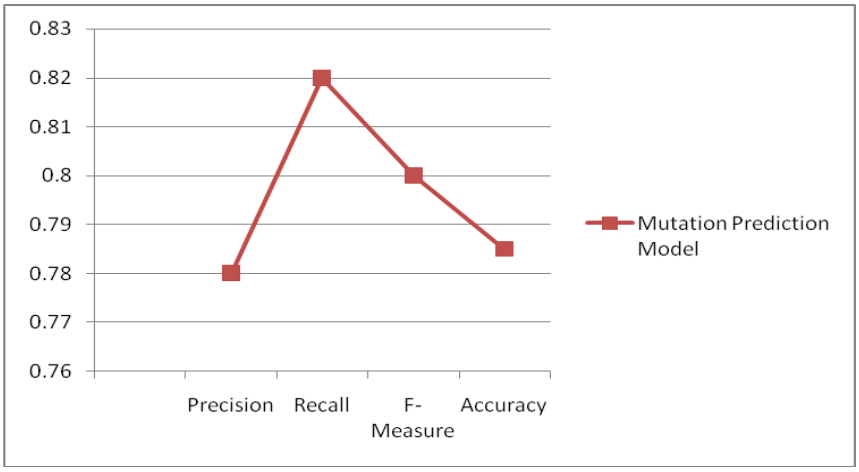


**Fig. 7.5 Performance of BRNN Based Mutation Prediction Model**

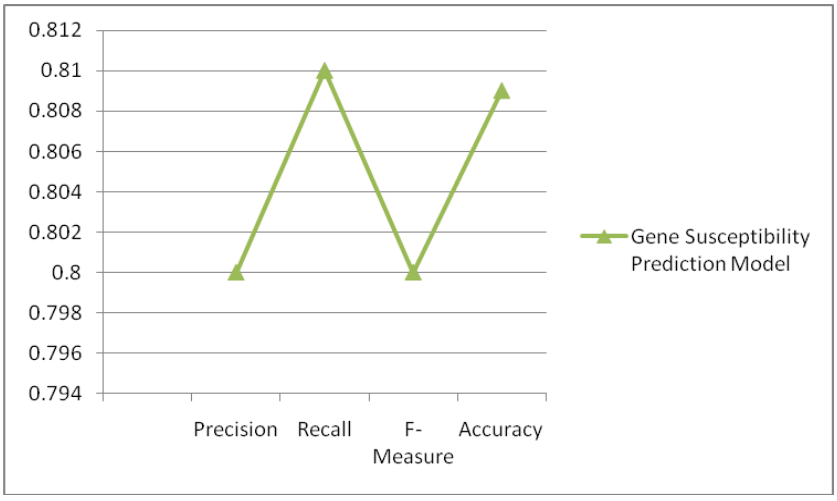


**Fig. 7.6 Performance of BRNN Based Gene Susceptibility Prediction Model**

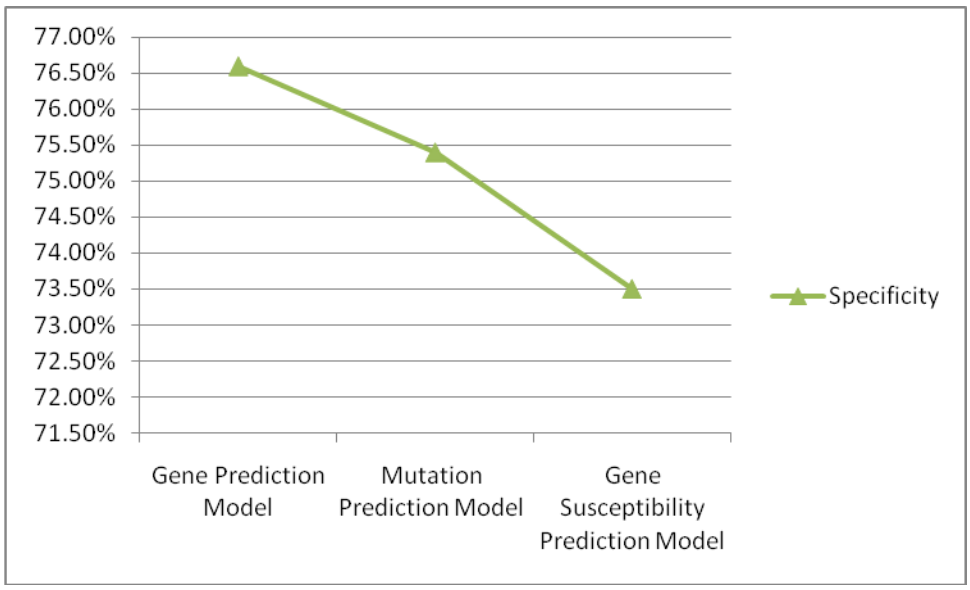


**Fig. 7.7 Specificity of BRNN Models**

As Fig.7.2 depicts, at the early epochs BRNN model achieves less accuracy, however, it achieves more accuracy in further epochs. Fig.7.3 clearly portrays that, the BRNN algorithm has soaring logarithmic loss in the initial epochs for all the three datasets but started decreasing after 100 epochs and reached the least in 250 epochs. The accuracy and precision of the BRNN model is high for ASD gene identification whereas recall value is high for mutation recognition task as depicted in Fig.7.4. In BRNN model when predicting gene susceptibility the specificity decreases which is illustrated in Fig. 7.7.

**Comparison of BRNN Model with DNN Model**

The effectiveness of the BRNN classifiers in predicting gene type, mutation category and gene susceptibility class is compared with the corresponding DNN models developed in the previous experiments described in chapter 6. The performance measures like precision, recall, accuracy and F-measure are used to compare BRNN with DNN and the comparative results is summarized in Table XXV.

**Table XXV Comparative Performance of BRNN and DNN Based Models**

| Metrics | Gene Prediction Model | | Mutation Prediction Model | | Gene Susceptibility Prediction Model | |
|---|---|---|---|---|---|---|
| | DNN | BRNN | DNN | BRNN | DNN | BRNN |
| Precision | 0.78 | 0.80 | 0.77 | 0.78 | 0.79 | 0.80 |
| Recall | 0.76 | 0.77 | 0.81 | 0.82 | 0.81 | 0.81 |
| F- Measure | 0.77 | 0.78 | 0.79 | 0.80 | 0.80 | 0.80 |
| Accuracy | 80.8% | 81.3% | 78.1% | 78.5% | 80.4% | 80.9% |

The tabulated results show that BRNN model consistently outperforms Deep Neural Network in all the three tasks. The precision and recall of BRNN model for the task of ASD gene classification is 0.80 and 0.77 which is slightly better than that of DNN with a precision of 0.78 and recall of 0.76. In the case of mutation identification, both the models show subtle differences and BRNN outperforms with an accuracy of 78.5%. The Bidirectional RNN is equally effective than DNN in classifying the ASD gene susceptibility with 80.9% accuracy which is 0.5% higher than that of DNN. The comparative performance of BRNN model with DNN for various measures of precision, recall, accuracy, F-measure is depicted in Fig.7.8.

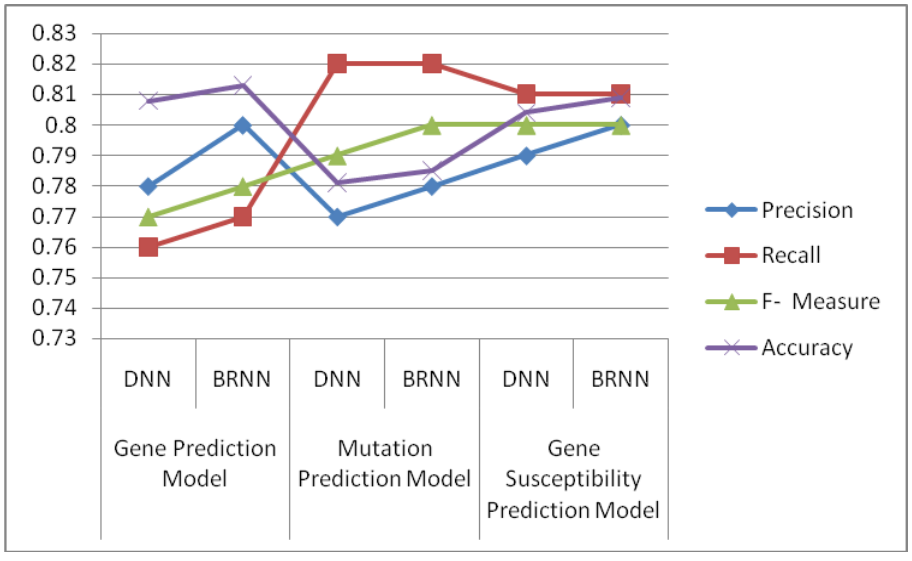


**Fig. 7.8 Comparative Performance of BRNN and DNN Based Models**

Fig.7.8 depicts the precision, recall, F-measure and accuracy values of BRNN and DNN models. It is noticed that BRNN outperforms DNN model in these metrics for the gene prediction model. The recall values are almost constant for both BRNN and DNN models for the mutation prediction and gene susceptibility prediction tasks. When considering the accuracy of the three prediction tasks, BRNN based models show constant increase for all these tasks.

**Findings**

The benefit of deep architecture based on bidirectional recurrent network to perform prediction of ASD gene, mutation and gene susceptibility is confirmed. It is evident that the performance of BRNN is relatively higher than DNN in all three tasks with respect to various measures as BRNN captures both past and future information data in both directions with two separate hidden layers. RNN has the capability of feedback loop which acts as a kind of memory to remember the past context enabling improved recognition rate. Also the BRNN network has learnt representations by integrating past, future information of the input to better understand the context, capture the high-level abstractions of the data and eliminate ambiguity to attain better performance. The hyperparameters tuned also add to the ability of the model and hence shows strong robustness in the prediction tasks. It is ascertained that the architectural potential of BRNN and the contributive features related to codon measures contribute to the overall performance in identifying ASD causing genes.

## 7.2. LSTM BASED MODEL TO PREDICT ASD GENES, THEIR SUSCEPTIBILITY AND MUTATIONS

Recurrent Neural Network architecture experiences drawback due to the diminishing effect of the inputs, which are farther into the past, causing short term memory effect. The standard RNN model is not capable to produce the most accurate results as it is able to look back a few timesteps only, which is not enough when more parameters are added. Also during updation of weights in RNN, there is instability in the gradients, introducing the problem of vanishing gradient when they happen to be very small number. To tackle this crisis and to achieve better results, an RNN variant, Long Short Term Memory network is employed in this work. LSTMs can forget less contributive features and remember influential patterns selectively for a long duration of time and hence can be used to create large recurrent networks, which in

turn can be used to address complex problems involving bio - sequence data [100]. This section describes the construction of stacked LSTM models to predict the ASD causing genes, their predisposition to ASD and mutations.

**Methodology**

The proposed LSTM prediction models is made up of three functional parts namely i) input component ii) process component iii) output component. The input component involves generation of mutation induced diseased ASD gene sequences, preparation of corpus and development of the dataset. The second component deals with building the three independent LSTM models for the relevant tasks. In the output component testing and performance evaluation of the models is dealt. The architecture of the proposed model is depicted in Fig.7.9.

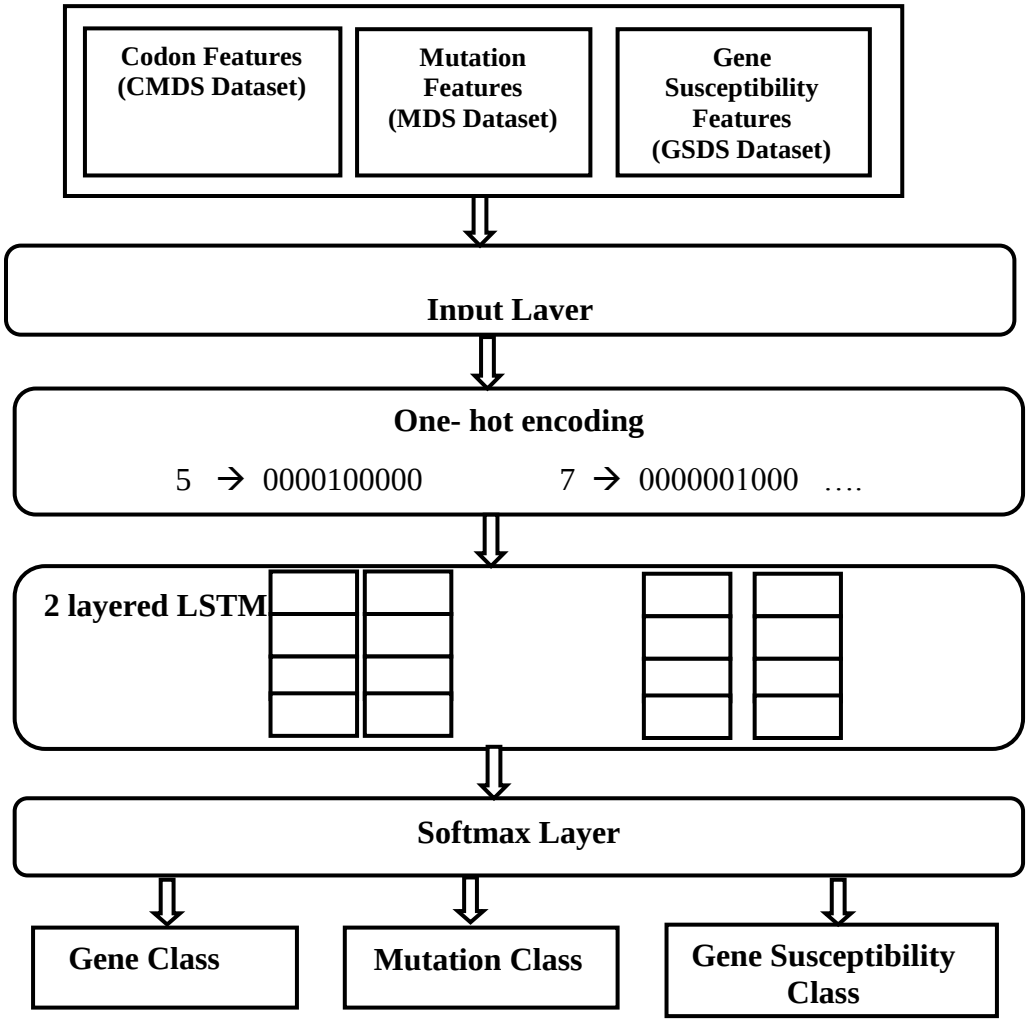


**Fig.7.9 Architecture of the LSTM Based Prediction Model**

In the first phase, the corpus developed using 1000 mutated gene sequences accounting for ten types of ASD genes and four types of mutations described in Chapter 3 is used. The three datasets CMDS, MDS and GSDS described in Chapter 4 and 5 are used to build the models for classification of genes, mutations and susceptibility prediction. One hot encoding of the class values is used where each value is represented by a binary vector. The feature vectors are reshaped into a format that can be used as input to the LSTM.

The second phase involves the construction of LSTM model which consists of one input layer, 2 hidden layers with 8 memory units and an output layer. Although BRNN used in the previous work integrates information from the backward and forward pass, it suffers from the problem of vanishing gradient and also cannot remember information for long contexts. It does not contain the facility to eliminate superfluous information. An LSTM network consists of memory cell which is smarter than a classic neuron and consists of three gates namely input, forget and output gate. Information is added to the cell state of LSTM by the input gate enabling the network to retain important information. Removal of information which are no longer required by LSTM is done by the forget gate. It also facilitates the network from being controlled by inputs which are very long in the past. The output gate modulates whether the output of the linear unit is to be broadcasted to the network. Two such LSTM units are stacked and the output of one unit is presented as input to the other. The output layer is a fully connected dense layer with as many neurons for the possible integers that may be output. In the ASD gene classification problem, the model consists of 10 neurons in the output layer and in the mutation classification model there are 4 neurons. The output layer consists of 3 neurons for predicting gene susceptibility to ASD. Softmax activation function is used in the output layer that allows the network to learn and output the probable class values. To improve the accuracy and efficiency of the two - layered LSTM architecture, various hyperparameters such as batch size, epochs, dropout, learning rate and optimizer are considered and are fine tuned. Three independent classifiers for gene identification, mutation recognition and gene susceptibility prediction have been built by training the LSTM network with three datasets.

In the output phase the models are tested for their performance using 10 - fold cross validation and evaluated using various evaluation measures such as precision, recall, F - measure, accuracy, log loss, specificity.

**Experiment and Results**

Experiments have been carried out by implementing LSTM network using Keras which is a high-level API for neural networks. The predictive models are built using a stacked two layered LSTM architecture wherein the output of one layer is further sent as input to the next layer. The network used a dropout rate of 0.3 and a learning rate of 0.01. When training and testing, data were segmented on mini-batches of size 64 data segments. The experiment was conducted for 250 epochs and the results are tabulated. The network used the sparse categorical cross entropy loss function while training, suitable for multiclass classification problems and the efficient Adam optimization algorithm. The LSTM is trained with the above parameter settings using CMDS dataset and the ASD causative gene identification model is built whereas the MDS dataset is used to train the mutation prediction model. Similar experiment is carried out on the GSDS dataset and the gene susceptibility recognition model is built. The performance of the models is evaluated on metrics like precison, recall, F-measure, accuracy and log loss to obtain insights about the suitability of these approaches for the objectives. The epochwise accuracy of these three models is shown in Table XXVI.

**Table XXVI Epochwise Accuracy of LSTM Models**

| Epochs | Gene Prediction Model | Mutation Prediction Model | Gene Susceptibility Prediction Model |
|---|---|---|---|
| 50 | 77.1% | 76.5% | 75.7% |
| 100 | 78.2% | 77.8% | 78.1% |
| 150 | 80.7% | 78.0% | 79.5% |
| 200 | 81.3% | 78.8% | 80.1% |
| 250 | 81.9% | 80.4% | 82.4% |

From the results in Table XXXI, it is seen that LSTM offers performance improvement for gene susceptibility classification when compared to the other tasks. The model achieves an accuracy of 82.4% for this task which is higher than the other two tasks. The experiments are conducted for epochs 50, 100, 150, 200, 250 and the accuracy of all three models is seen to increase gradually with the increase in the epochs. The gene prediction model attains 77.1% at

50 epochs and reaches 81.9% at 250 epochs. Initially at 50 epochs, the mutation prediction model accomplishes 76.5% and later at 250 epochs arrives at 80.4%. The accuracy of gene susceptibility prediction model starts with 75.7% in the initial epoch reaches 80.1% in 200 epochs and finally attains 82.4% at 250 epochs. The epochwise log loss of these three models is shown in Table XXVII.

**Table XXVII Epochwise Log loss of LSTM Models**

| Epochs | Gene Prediction Model | Mutation Prediction Model | Gene Susceptibility Prediction Model |
|---|---|---|---|
| 50 | 0.9102 | 1.2197 | 1.7952 |
| 100 | 0.8995 | 1.2842 | 1.4614 |
| 150 | 0.8482 | 1.1713 | 1.2830 |
| 200 | 0.8617 | 1.0165 | 1.0489 |
| 250 | 0.8284 | 0.9562 | 0.7811 |

From the results it is seen that the log loss associated with classifying the predisposition of genes to the disorder is 0.7811 which is comparatively less when compared to that of 0.8284 for gene classification and 0.9562 for mutation recognition at 250 epochs. The gene susceptibility prediction model has an initial log loss of 1.7952 whereas the gene prediction and mutation prediction models have 0.9102 and 1.2197 loss at 50 epochs. It is observed that the log loss reduces progressively as epochs increase and the models have achieved better performance by reducing its misclassifications. A steady decrease in the log loss of gene susceptibility prediction model is noticed from 50 to 250 epochs. The performance comparison of these three models is shown in Table XXVIII.

**Table XXVIII Performance Results of LSTM Models**

| Metrics | Gene Prediction Model | Mutation Prediction Model | Gene Susceptibility Prediction Model |
|---|---|---|---|
| Precision | 0.81 | 0.79 | 0.80 |
| Recall | 0.78 | 0.81 | 0.81 |
| F- Measure | 0.79 | 0.80 | 0.80 |
| Accuracy | 81.9% | 80.4% | 82.4% |
| Correctly classified instances | 409 | 396 | 414 |
| Incorrectly classified instances | 91 | 104 | 86 |
| Specificity | 73.8% | 76.8% | 77.3% |

The performance of the LSTM model is evaluated using precision, recall and F-measure. As depicted in the above table LSTM has an upper edge for the ASD gene susceptibility prediction model. It is effective in predicting the ASD gene susceptibility with a precision of 0.8, recall of 0.81 and F-measure of 0.80. The LSTM classifier performs fairly well with precision of 0.81 and recall of 0.78 for gene prediction model. The gene susceptibility prediction model using LSTM has correctly identified 413 instances. When evaluating the specificity, the susceptibility prediction gives a prominent score value of 77.3% whereas it is 73.8% and 76.8 % for ASD gene and mutation prediction models. The charts Fig.7.10 to Fig.7.13 depict the result analysis.

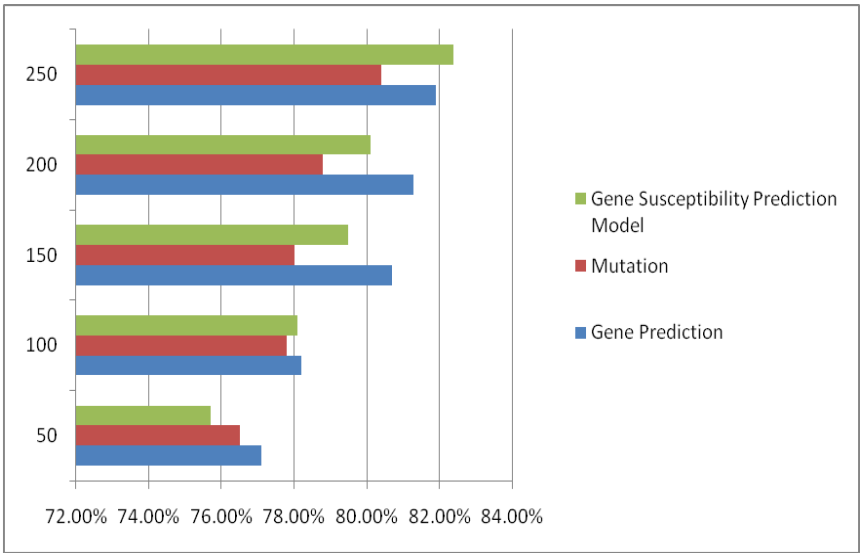


**Fig 7.10 Epochwise Accuracy of LSTM Models**

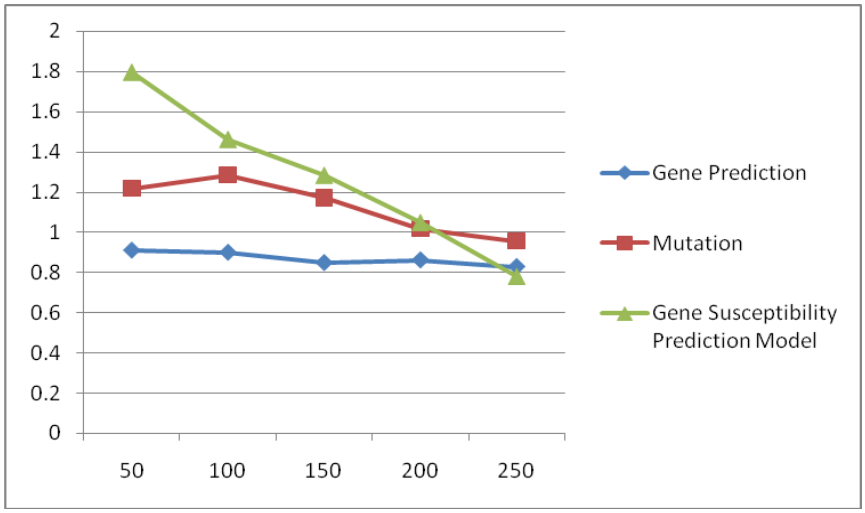


**Fig. 7.11 Epochwise Logloss of LSTM Models**

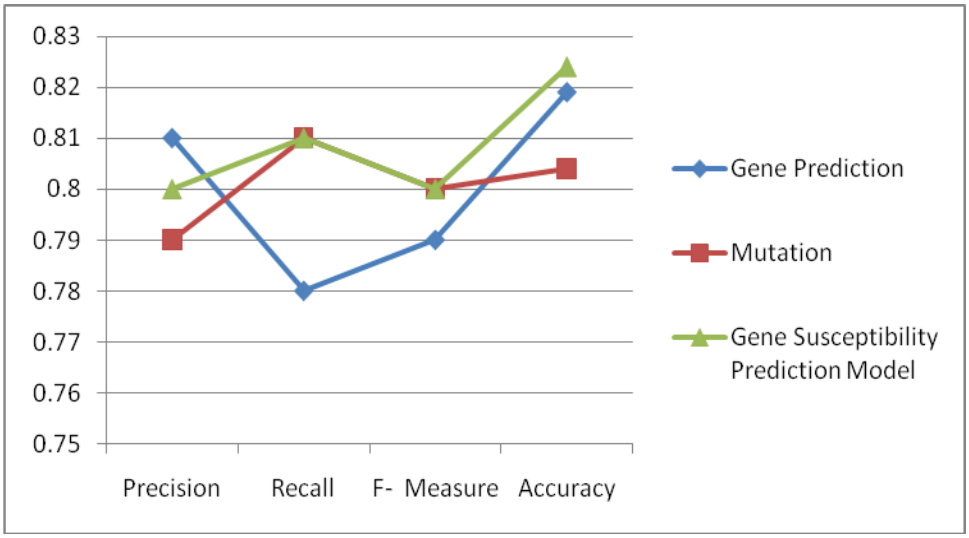


**Fig 7.12 Performance of LSTM Models**

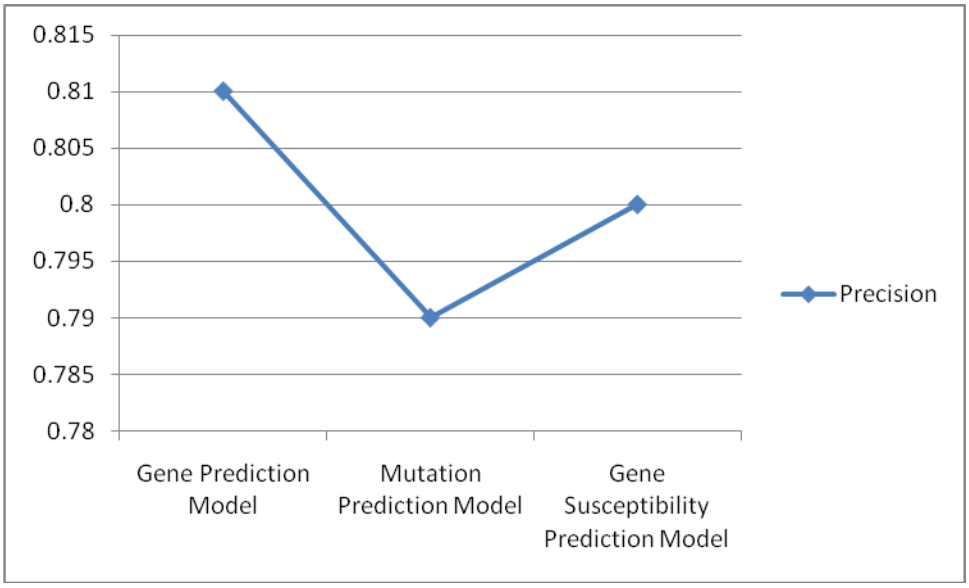


**Fig. 7.13 Precision of LSTM Models**

Fig.7.10 shows that the epochwise accuracy values are higher for LSTM based gene susceptibility prediction model. The logloss of the models is illustrated in Fig.7.11 which depicts that it is minimum with 0.7811 for the gene prediction model. In the case of mutation prediction model the logloss elevates at 250 epochs and then reaches 0.9562. It is noticed from Fig.7.12 that the precision and accuracy of LSTM based gene susceptibility prediction model is high whereas the recall value is almost similar to mutation prediction model. LSTM based gene susceptibility prediction model has more precision as shown in Fig.7.13.

**Comparison of LSTM with DNN and BRNN**

The performance of the LSTM based gene, mutation, susceptibility prediction models is compared with corresponding models of DNN and BRNN developed in the previous experiments. The DNN have shown better performance in predicting susceptibility of genes to ASD and the BRNN is promising in recognizing ASD causing genes. These models are compared with the LSTM model which effectively handles vanishing gradients and remembers context for a long period of time. The performance measures like precision, recall, accuracy and F-measures are used to compare LSTM with the other two deep learning models. The experiments have been conducted for multiple epochs but the values attained at 250 epochs are reported in Table XXIX.

**Table XXIX Comparative Performance of DNN, BRNN and LSTM Based Models**

| Metrics | Gene Prediction Model | | | Mutation Prediction Model | | | Gene Susceptibility Prediction Model | | |
|---|---|---|---|---|---|---|---|---|---|
| | DNN | BRNN | LSTM | DNN | BRNN | LSTM | DNN | BRNN | LSTM |
| Precision | 0.78 | 0.80 | 0.81 | 0.77 | 0.78 | 0.79 | 0.79 | 0.81 | 0.80 |
| Recall | 0.76 | 0.77 | 0.78 | 0.81 | 0.82 | 0.82 | 0.81 | 0.82 | 0.81 |
| F- Measure | 0.77 | 0.78 | 0.79 | 0.79 | 0.80 | 0.80 | 0.80 | 0.81 | 0.80 |
| Accuracy | 80.8% | 81.3% | 81.9% | 78.1% | 78.5% | 80.4% | 80.4% | 80.9% | 82.4% |

The results prove that LSTM model outperforms the other deep architectures in identifying the ASD causing genes and their susceptibility. LSTM based gene identification model shows superior accuracy of 81.91% whereas BRNN and DNN have achieved 81.3% and 80.8% for the same. The LSTM model performs comparatively better than DNN and BRNN to identify the ASD genes susceptibility with precision of 0.80 and recall of 0.81. In the case of mutation prediction model LSTM has an accuracy of 79.3 which is closer to 80.4% of BRNN. The comparative performance of LSTM model with other deep models for various measures of precision, recall, accuracy, F-measure is depicted in Fig.7.14.

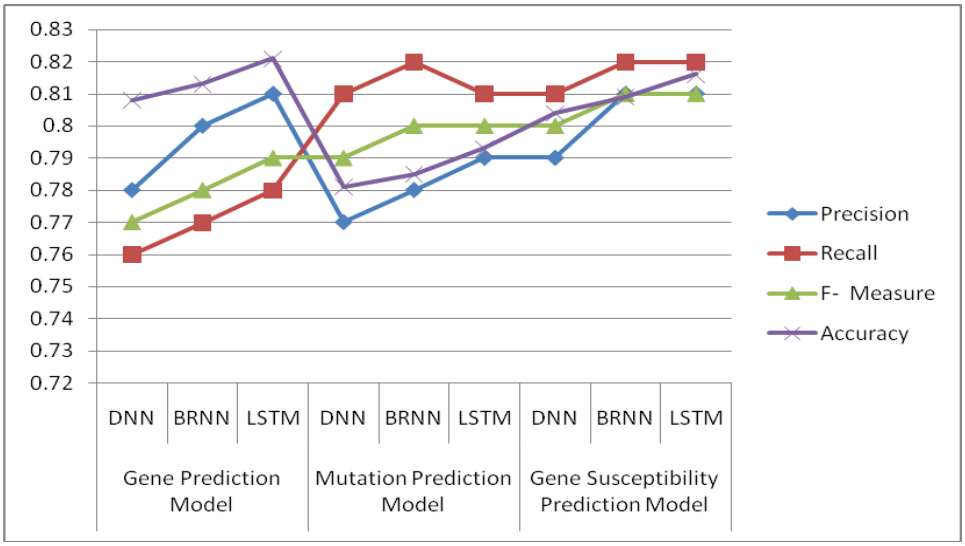


**Fig. 7.14 Comparative Performance of DNN, BRNN and LSTM Based Models**

The performance evaluation of LSTM models based on precision, recall and F- measure depicted in Fig.7.14 show that LSTM performs convincingly better than other deep learning models. The comparative performance of DNN, BRNN and LSTM illustrate that the accuracy of gene prediction model attains a peak at 0.82 for LSTM model whereas for DNN and BRNN it is 80.8% and 81.3% respectively. The recall values for the BRNN and LSTM model reaches a maximum of 0.82 for the gene susceptibility prediction model. The precision values are constantly increasing for all three models but LSTM stands out with a maximum of 0.81.

**Findings**

The empirical results confirm that the LSTM model developed for predicting ASD gene, mutation and gene susceptibility works with high rate of recognition. It is apparent that the stacked two layered LSTM architecture performs relatively better than DNN and BRNN in recognizing the ASD genes and their susceptibility as the network is able to remember the context for very long durations. LSTM network, modulated by the state of the cells, offers better prediction based on the historical context of inputs, rather than only on the very last input. This stacked LSTM network updates only the necessary details and small modifications to the information are done by multiplications and additions which decrease the computational complexity and eventually the time complexity.

It is established that this hierarchy of hidden layers in the stacked LSTM has helped in complex representation of the data, capturing information at different scales which adds value to the results. The capability of the network to selectively remember contributive features and forgets things which are not prominent enables in improved prediction. The proposed LSTM model is promising in identification of ASD gene susceptibility as the precision, recall and F - measure of the LSTM based models is high. Also it is fairly performing good for gene and mutation prediction task.

## 7.3. GRU BASED MODEL TO PREDICT ASD GENES, THEIR SUSCEPTIBILITY AND MUTATIONS

GRU, a variant of Recurrent Neural Networks (RNNs), have shown to achieve state-of-the-art results in many applications with sequential data, including machine translation and speech recognition. GRUs have the ability to retain memory from previous activations rather than replacing the entire activation like a standard RNN, allowing them to remember features for a long time. It also lets backpropagation to happen through multiple bounded nonlinearities, which reduces the likelihood of the vanishing gradient. In GRUs the gating mechanism provide better control in the flow of the information through the various time steps [101]. GRU is faster to train than any other deep learning models and needs fewer data to generalize. To incorporate the benefits of GRU architecture, it is proposed to apply it in the next level of research for the identification of ASD genes, mutations and gene susceptibility. Customized GRU model is attempted here as it is a new promising venue to build effective prediction models for further enhancement.

### Methodology

In this work GRU network exploiting the shared feature extraction between user defined layers is employed to distinguish ASD causing genes, their susceptibility and driving mutations. The sequential method of creating deep architectures contains layer-by-layer model and does not allow creation of models that share layers or have multiple inputs or outputs. Alternatively, the functional models have a lot more flexibility and can be easily defined where layers connect to more than just the previous and next layers. The GRU based functional model is defined by creating instances of layers and connecting them directly to each other in pairs. The proposed

methodology includes three functional parts such as datasets creation, model building and performance evaluation and is depicted in Fig.7.15.

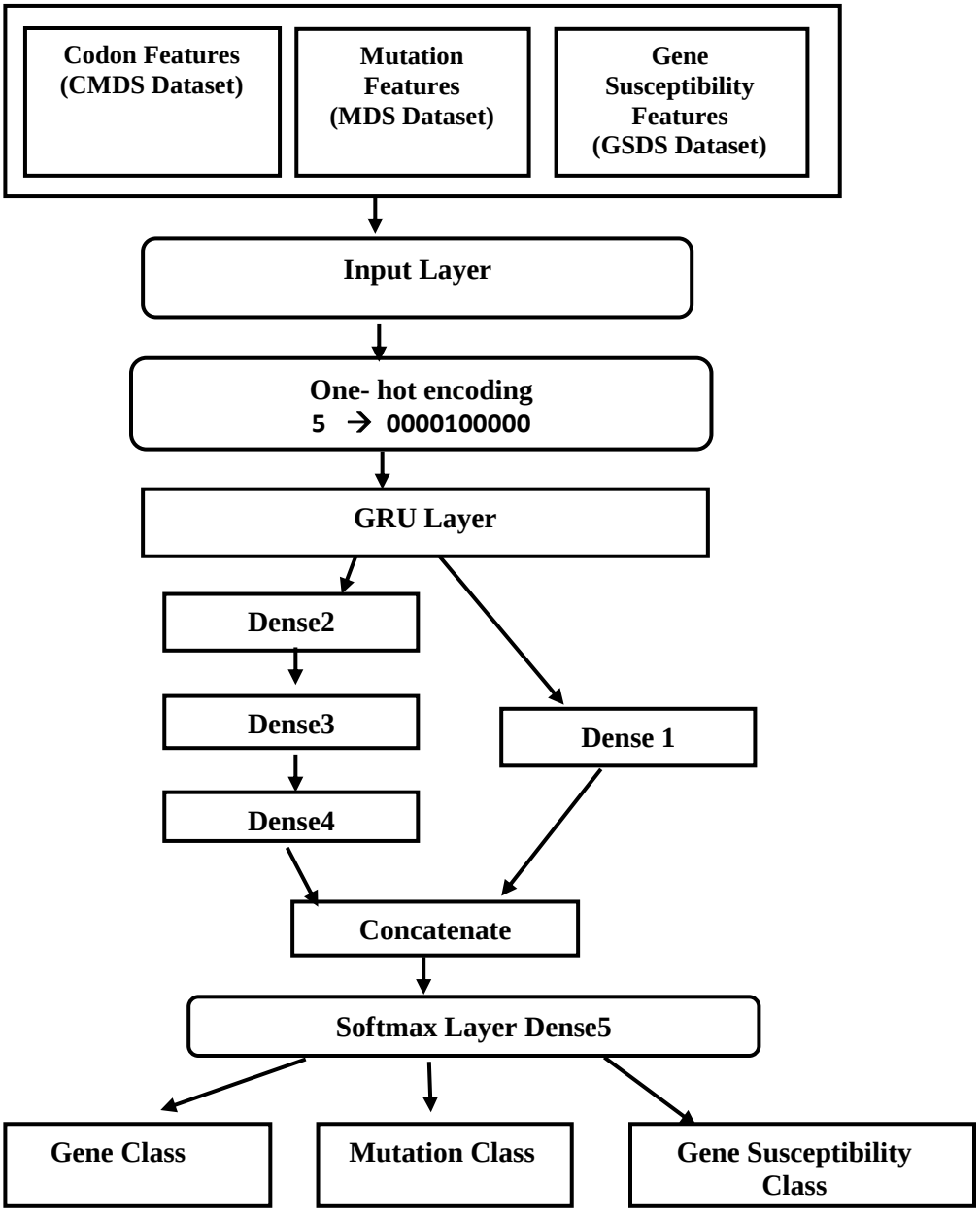


**Fig. 7.15 Architecture of GRU Based Prediction Model**

In the first phase, the corpus developed using 1000 mutated gene sequences accounting for ten types of ASD genes and four types of mutations described in Chapter 3 is used. The three datasets CMDS, MDS and GSDS described in Chapter 4 and 5 are used to build the models for classification of genes, mutations and susceptibility prediction. One hot encoding of the class

values is used where each value is represented by a binary vector. The feature vectors are reshaped into a format that can be used as input to the GRU.

In the second phase a user defined GRU model is designed with input and output layers with two submodels that share layers. In this work, there are two parallel submodels designed to interpret the output of a GRU feature extractor. The input to the model is fed to the GRU layer with 8 memory cells which interprets the data. In this model the layers take a more functional form as compared to the sequential model. The inputs to each layer are explicitly specified and the output of each layer is controlled. The first layer passes input to GRU layer which in turn acts as input to dense_1 and dense_2. Further Dense_3 receives dense_2 as input and is passed on to dense_4. Now the output from both dense_1 and dense_4 is concatenated and given as input to dense_5. This method of passing the inputs to the next layer allows the tensors to be shared with multiple layers. The trainable weights from the user defined layers are given to the dense layers with the purpose of fine tuning the parameters and reducing the learning time. The first interpretation model is a shallow single fully connected layer and the second is a deep 3 layer model. The output of both interpretation models are concatenated into one long vector which is passed to the output layer used for multiclass classification. The output layer is a fully connected dense layer with n neurons for the n possible integers that may be output. There are 10 neurons in the output layer of the ASD gene classification model and in the mutation classification model the output layer has 4 neurons. The output layer consists of 3 neurons for predicting gene susceptibility to ASD.

A softmax activation function is used on the output layer to allow the network to learn and calculate the probabilities of each gene type over all possible targets. The predictive model is constructed with sparse categorical cross entropy loss function for training and suitable for prediction problems. Here various hyperparameters such as batch size, epochs, dropout, learning rate and optimizer are considered and fine tuned to improve the model accuracy. The parameter dropout is tuned by selecting the nodes randomly which are to be dropped out with a given probability during updation of weights. Learning rate is set to contrive the weights within the direction opposite of the gradient for given input values. Optimization algorithm calculates an exponential moving average of the gradient and the squared gradient. The purpose of the optimization algorithm is to minimize or maximize an objective function with respect to internal

parameters of the model. Three independent classifiers for gene identification, mutation recognition and gene susceptibility prediction have been built by training the GRU network with three datasets.

Eventually the three models are tested using 10 - fold cross validation and their predictive performance is evaluated using various metrics such as precision, recall, F- measure, accuracy and log loss.

**Experiment and Results**

Experiments have been carried out by implementing GRU with shared feature extraction between layers using the Keras functional API. The Keras functional API is useful for creating complex models, such as multi-input / multi-output models, directed acyclic graphs and models with shared layers. GRU layer is implemented by setting the various hyperparameters as mentioned in chapter 6. For the sake of efficiency, when training and testing, data were segmented on mini-batches of size 64 data segments. The learning rate of 0.01 is used and when varying dropouts from 0.2 to 0.5 were experimented it was found that dropout of 0.3 was optimal. The Adam optimization algorithm was used and epochs of 50, 100, 150, 200, 250 are experimented. Three models are built under these parameter settings. The GRU model is trained using these parameter settings with CMDS dataset and the ASD causative gene identification model is built. The MDS dataset is used to train the mutation prediction model and similar experiment is carried out on the GSDS dataset to build the gene susceptibility recognition model.

The standard 10 - fold cross-validation technique is applied and the performance of the gene prediction, mutation identification and gene susceptibility recognition models were evaluated based on prediction accuracy, logarithmic loss, precision, recall and F-measure. The prediction results of the models are tabulated. The epochwise accuracy of these three models is shown in Table XXX.

**Table XXX Epochwise Accuracy of GRU for Three Models**

| Epochs | Gene Prediction Model | Mutation Prediction Model | Gene Susceptibility Prediction Model |
|---|---|---|---|
| 50 | 78.7% | 80.1% | 77.8% |
| 100 | 79.2% | 80.6% | 78.1% |
| 150 | 81.5% | 81.0% | 78.7% |
| 200 | 82.0% | 81.5% | 79.2% |
| 250 | 82.5% | 81.8% | 80% |

The tabulated results show that GRU based ASD gene prediction model has achieved an accuracy of 82.5% at 250 epochs which is higher than that of gene susceptibility identification model which is 80.0%. At 50 epochs the mutation prediction model achieved an accuracy of 80.1%, gradually increased to 81% at 150 epochs and reached 81.8% at 250 epochs. There is an increase of about 2.2% accuracy for the gene susceptibility prediction model from 50 to 250 epochs. The accuracy of gene prediction model at 50 epochs is 78.7% , 81.5% at 150 epochs and attains a maximum of 82.5% at 250 epochs. The experiments prove that accuracy of all three models increases as epochs are increased. The epochwise log loss of these three models is shown in Table XXXI.

**Table XXXI Epochwise Logloss of GRU Models**

| Epochs | Gene Prediction Model | Mutation Prediction Model | Gene Susceptibility Prediction Model |
|---|---|---|---|
| 50 | 0.9845 | 0.9523 | 1.0489 |
| 100 | 0.9661 | 0.9515 | 0.9961 |
| 150 | 0.8015 | 0.8893 | 0.9215 |
| 200 | 0.7965 | 0.8417 | 0.8268 |
| 250 | 0.7162 | 0.8184 | 0.7615 |

The results illustrate that the log loss reduces as epochs increase and the models have achieved better performance by reducing its misclassifications. Initially at 50 epochs the three models had log loss of 0.9845, 0.953 and 1.0849 for predicting genes, mutations and susceptibility respectively. This gets reduced with a difference of 0.2683, 0.1339 and 0.2874

respectively for the above three models at the end of 250 epochs. The log loss associated with classifying the ASD causative genes is 0.7162 which is comparatively less when compared to that of 0.8184 for mutation classification and 0.7615 for recognizing the predisposition of genes to the disorder. The performance comparison of these three models is shown in Table XXXII.

**Table XXXII Performance Results of GRU Based Models**

| Metrics | Gene Prediction Model | Mutation Prediction Model | Gene Susceptibility Prediction Model |
|---|---|---|---|
| Precision | 0.83 | 0.81 | 0.79 |
| Recall | 0.81 | 0.82 | 0.81 |
| F- Measure | 0.82 | 0.81 | 0.8 |
| Accuracy | 82.5% | 81.8% | 80.00% |
| Correctly classified instances | 414 | 409 | 401 |
| Incorrectly classified instances | 86 | 91 | 99 |
| Specificity | 80.3% | 78.3% | 77.4% |

The result analysis indicates that GRU model shows promising performance for the ASD gene prediction model. It is effective in predicting the ASD genes with a precision of 0.83, recall of 0.81 and F-measure of 0.82. The DNN classifier achieves a precision of 0.81 and recall of 0.82 for mutation prediction model. The GRU based gene susceptibility prediction model has correctly identified 401 instances whereas for the mutation prediction it has identified 409 instances with 0.81 precision and a recall of 0.82. When evaluating the specificity, GRU gives a prominent score value of 80.3% for identifying the genes whereas it is 78.3% and 77.4% for mutation and susceptibility prediction models. The experimental results of GRU based models are illustrated in Fig.7.16 to Fig.7.20.

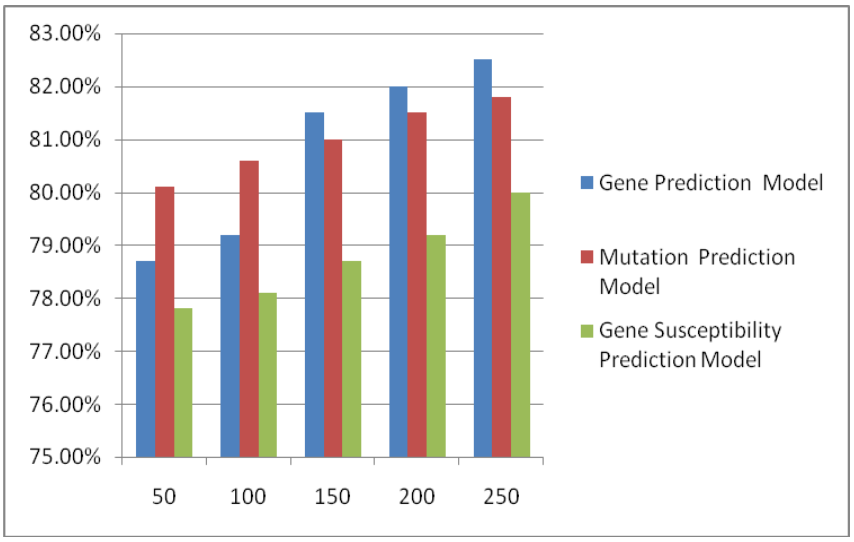


**Fig. 7.16 Epochwise Accuracy of GRU Based Models**

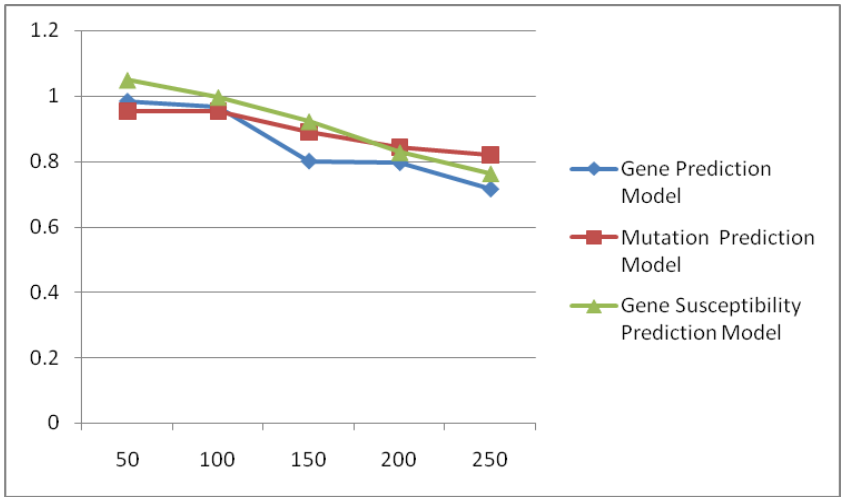


**Fig. 7.17 Epochwise Log Loss of GRU Based Models**

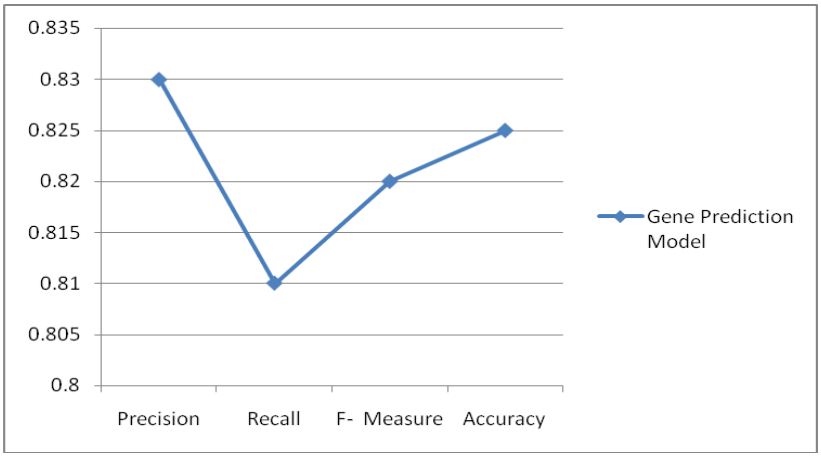


**Fig. 7.18 Performance of GRU Based Gene Prediction Model**

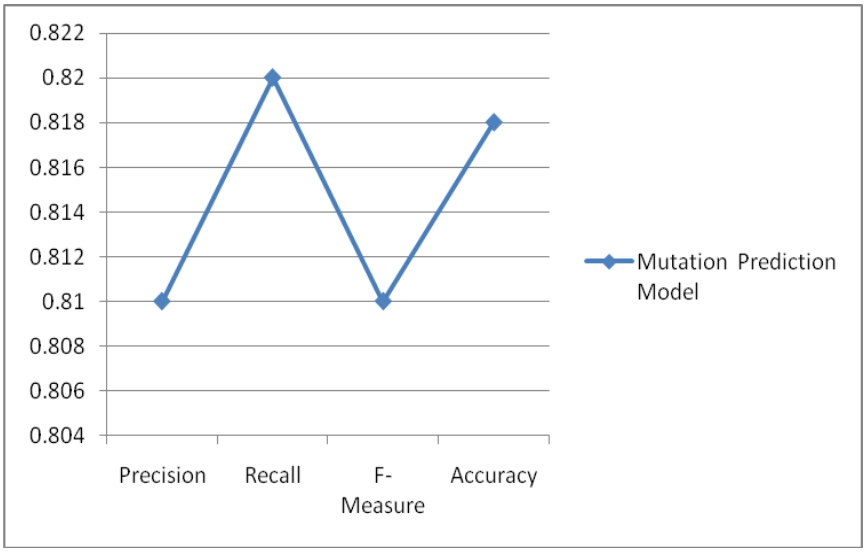


**Fig. 7.19 Performance of GRU Based Mutation Prediction Model**

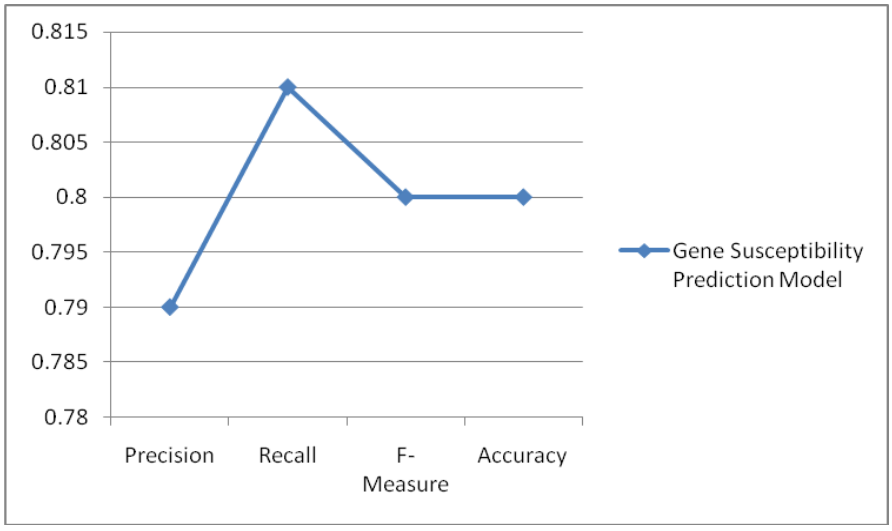


**Fig. 7.20 Performance of GRU Based Gene Susceptibility Prediction Model**

Fig.7.16 shows that the epochwise accuracy values are higher for GRU based gene recognition model. The logloss which is initially high for all models reduces with increased epochs and is observed to be minimum for the gene prediction model which is illustrated in Fig.7.17. In the case of Mutation Prediction Model the logloss elevates at 100 epochs and then reduces whereas the gene susceptibility prediction model has a logloss of 0.8159 at 250 epochs. It is noticed from Fig.7.18 and Fig.7.19 that the precision of gene prediction model is high

whereas the recall has a dip of 0.01 compared to mutation prediction model. The recall value of gene susceptibility prediction model is high with 0.82 as shown in Fig.7.20.

**Comparison of GRU with DNN, BRNN, LSTM**

The effectiveness of the GRU classifier is compared with other deep learning models developed using DNN, BRNN and LSTM in the previous experiments. The performance measures like precision, recall, accuracy and F-measures are used to compare GRU based models with the other deep network architectures. The comparative results of the GRU based ASD gene prediction model with other deep models is reported in Table XXXIII. Table XXXIV depicts the comparative results of the GRU based mutation prediction model with other deep models. In Table XXXV the results of the GRU based gene susceptibility prediction model is compared with that of other deep models.

**Table XXXIII Comparative Results of GRU Based ASD Gene Prediction Model with other Deep Models**

| Metrics | DNN | BRNN | LSTM | GRU |
|---|---|---|---|---|
| Precision | 0.78 | 0.80 | 0.81 | 0.83 |
| Recall | 0.76 | 0.77 | 0.78 | 0.81 |
| F- Measure | 0.77 | 0.78 | 0.79 | 0.82 |
| Accuracy | 80.8% | 81.3% | 81.9% | 82.5% |

**Table XXXIV Comparative Results of GRU Based ASD Mutation Prediction Model with other Deep Models**

| Metrics | DNN | BRNN | LSTM | GRU |
|---|---|---|---|---|
| Precision | 0.77 | 0.78 | 0.79 | 0.81 |
| Recall | 0.81 | 0.82 | 0.81 | 0.82 |
| F- Measure | 0.79 | 0.80 | 0.80 | 0.81 |
| Accuracy | 78.1% | 78.5% | 80.4% | 81.8% |

**Table XXXV Comparative Results of GRU Based ASD Gene Susceptibility Prediction Model with other Deep Models**

| Metrics | DNN | BRNN | LSTM | GRU |
|---|---|---|---|---|
| Precision | 0.79 | 0.81 | 0.80 | 0.79 |
| Recall | 0.81 | 0.82 | 0.81 | 0.81 |
| F- Measure | 0.80 | 0.81 | 0.80 | 0.8 |
| Accuracy | 80.4% | 80.9% | 82.4% | 80.5% |

The comparative results in Table XXXIII prove that GRU model outperforms the other three deep models to identify the ASD causing genes. The LSTM model achieved precision of 0.81 and recall of 0.78 whereas GRU has achieved 0.83 and 0.82 for the same task. As shown in Table XXXIV, the GRU model performs comparatively better than DNN, BRNN and LSTM models to identify the ASD triggering mutations with precision of 0.81 and recall of 0.82. In the case of gene susceptibility prediction model LSTM outperforms GRU as its accuracy is 82.4% which is 2% less than LSTM model as shown in Table XXXV. The comparative results of log loss of GRU based prediction model with other deep models is shown in Table XXXVI.

**Table XXXVI Comparative Results of Log Loss of GRU Based Prediction Models with other Deep Models**

| Architecture | Gene Prediction Model | Mutation Prediction Model | Gene Susceptibility Prediction Model |
|---|---|---|---|
| DNN | 0.8641 | 0.8415 | 0.8159 |
| BRNN | 0.8010 | 0.9615 | 0.9459 |
| LSTM | 0.8284 | 0.9562 | 0.7811 |
| GRU | 0.7162 | 0.8184 | 0.7615 |

As results depict, GRU model has comparatively less misclassifications for all three models and hence evidenced reduced logarithmic loss of 0.7162, 0.8184 and 0.7615 respectively. High log loss of 0.8641 was evidenced by the DNN based gene prediction model, 0.9615 by BRNN based mutation prediction model and 0.9459 by BRNN based gene susceptibility prediction model. The comparative performance of GRU model with other deep models for various measures of precision, recall, accuracy, F-measure and log loss is depicted in Fig.7.21 – Fig.7.24.

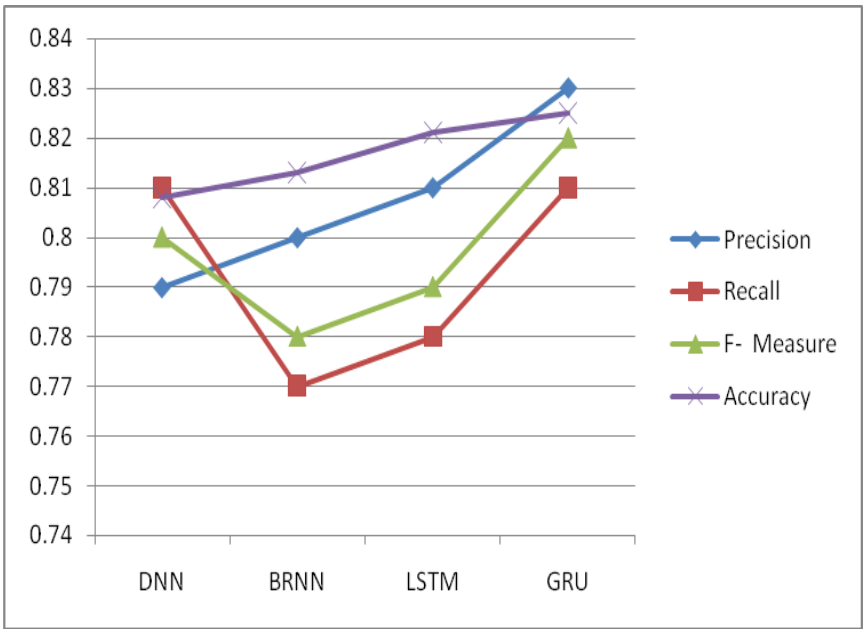


**Fig. 7.21 Comparative Performance of GRU Based ASD Gene Prediction Model with other Deep Models**

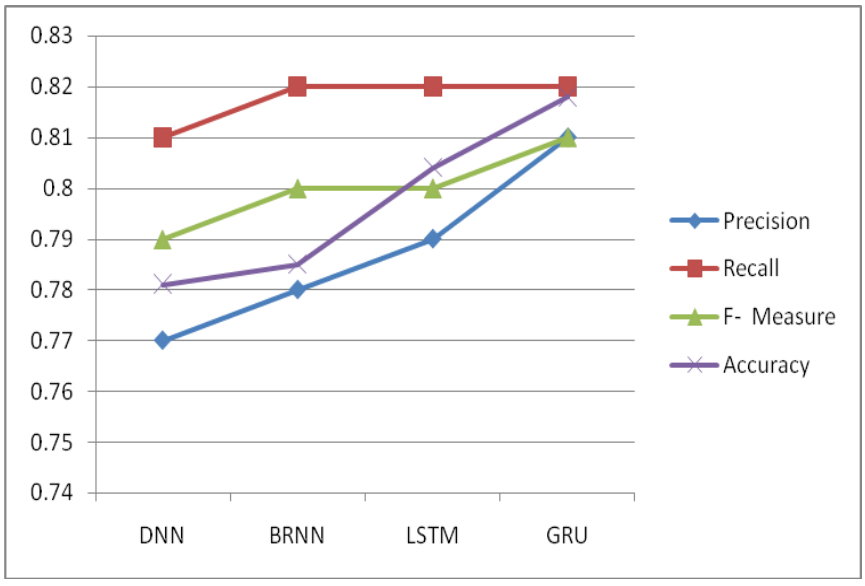


**Fig. 7.22 Comparative Performance of GRU Based ASD Mutation Prediction Model with other Deep Models**

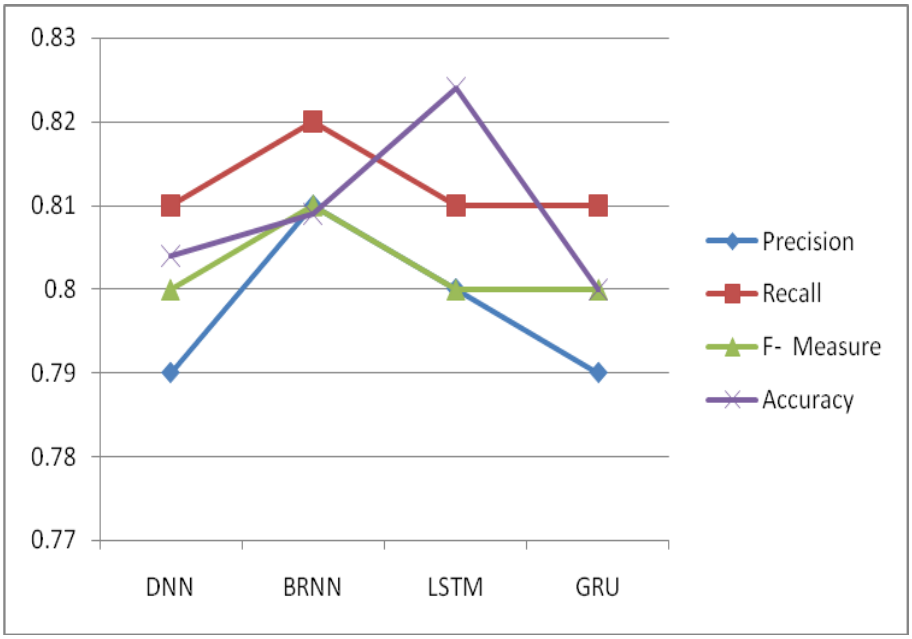


**Fig. 7.23 Comparative Performance of GRU Based ASD Gene Susceptibility Prediction Model with other Deep Models**

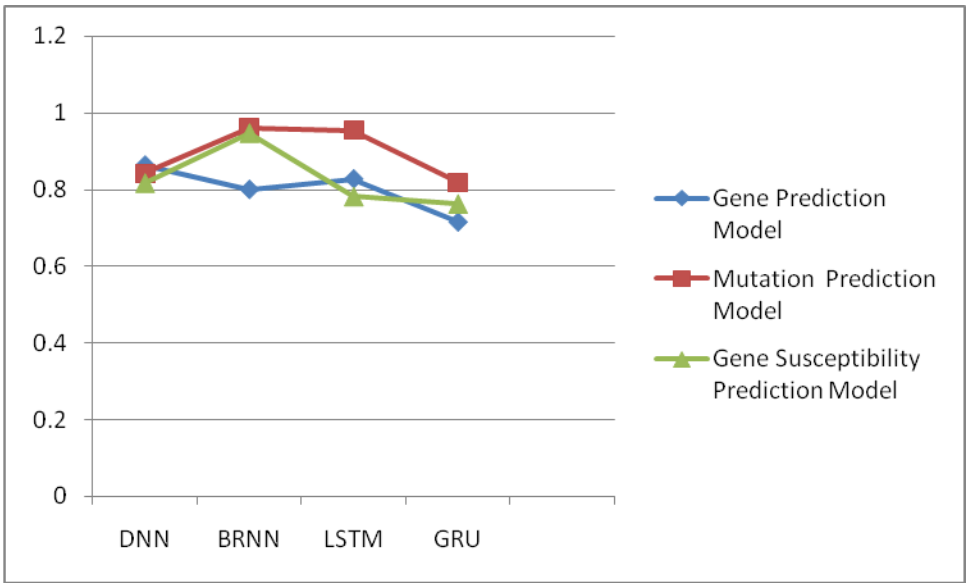


**Fig. 7.24 Log Loss Comparison of GRU Based Prediction Models with other Deep Models**

The result analysis indicate that the LSTM and GRU models have almost equal accuracy in predicting the genes but GRU outperforms with high precision and recall than LSTM as shown in Fig.7.23. With regard to mutation prediction, the recall values of BRNN, LSTM and GRU appear the same whereas accuracy is constantly soaring for GRU than the other models and is shown in Fig.7.24. The LSTM based gene susceptibility prediction model performs slightly better than GRU based model with 0.2 differences in precision and accuracy which is portrayed in Fig.7.25. The performance of the models with regard to log loss is given in Fig.7.26 and it shows that the GRU based models has less log loss.

**Findings**

GRU model involves two sub models which are parallelly processed to learn and share the parameters thereby reducing the total number of parameters needed to learn. The features are jointly learned by the user defined models and integrated to capture the relevant features and improve the generalization. Both the layers have learned parameters which contributed significantly in the classification task. Compared to other deep models DNN, BRNN and LSTM, GRU has the advantage of less parameters and easier training. The proposed method also outperforms DNN, BRNN, LSTM due to less number of updations in update gate z and reset gate r. The loss associated with misclassifications is reduced for all three GRU based prediction models. The experiments of GRU architecture on three datasets and the empirical results ascertain that the prediction of ASD causative genes and mutations can be done efficiently using this model.

**SUMMARY**

Recurrent Neural Network based models using BRNN, LSTM and GRU developed to classify ASD gene sequences, their predisposition to ASD and driving mutations have been elaborated in this chapter. The models were trained with three different datasets and the effectiveness of these models was evaluated using different performance metrics to explore the reliability of the method. The results of the experiment were also demonstrated in this chapter using tables and charts. The experimental results of the BRNN based predictive model was compared with DNN model and reported. The models built using RNN variants, LSTM and

GRU were compared with DNN and BRNN models and the result analyses is given in this chapter. The interpretations of the experimental results were also presented.

**Remarks**

- Paper titled "Bidirectional Recurrent Neural Network Based Model to Identify ASD", accepted for publication in the Journal of Advances in Modelling and Analysis A.**(Scopus indexed)**