CHAPTER

9

Exploring tree-based machine learning methods to predict autism spectrum disorder

Kazi Shahrukh Omar[a], *Muhammad Nazrul Islam*[b,*] *and Nabila Shahnaz Khan*[c]

[a]Department of Computer Science and Engineering, Uttara University, Dhaka, Bangladesh; [b]Department of Computer Science and Engineering, Military Institute of Science and Technology (MIST), Dhaka, Bangladesh; [c]Department of Computer Science, University of Central Florida, Orlando, FL, United States

**Corresponding author*

1 Introduction

Autism, or autism spectrum disorder (ASD), refers to a neurodevelopmental disorder that is characterized by a person's repetitive behaviors, social interaction and skills, nonverbal and speech communication, and learning skills. The symptoms of autism generally appear in childhood (within 2–5 years old) and then develop through time. Thus the diagnosis of autism can be done at any age: childhood, adolescence, or adulthood [1]. Several challenges are met by people with ASD which include difficulties with concentration; attention to details; unable to understand other perspectives; facing difficulties in learning; and mental health problems such as depression, anxiety, and the likes.

Several etiologies present in the biology of autism make it difficult to generalize [2]. For example, some children may have frank immune disorders, while others may bear the gastrointestinal disease. Additionally, in some cases autism signs become observable from birth, and for others the symptoms regress with time. Autism begins at early stage (childhood) and generally persists into adolescence and adulthood. According to Mayada et al. [3], in every 160 children, 1 child has ASD. People having ASD may live independently or may require consistent care and assistance depending on the severity of their disabilities.

Several screening tools exist primarily to screen autism in young toddlers, while some tools exist for matured persons, teenagers, and young children [4]. A number of researches have been conducted focusing on developing such screening tools as well as comparing the performance of such tools. Machine learning

(ML) plays a crucial role in diagnosing various diseases, including the diagnosis of autism or ASD [5]. Using sufficiently large ASD screening datasets, ML can predict autism traits up to a certain accuracy level in an individual and thus has opened a wide range of scope for further research in this field. A survey-based research was done by Hyde et al. [6] which not only reviewed existing research works in this field but also introduced possibilities of further research in this field using supervised ML. In Ref. [7], Lee et al. compared the performance of several ML techniques, including random forest, NB–SVM (Naïve Bayes–support vector machine), and NN, to predict ASD among the children in the United States. A comprehensive analysis was done by Bone et al. [8], where they focused both on the possibilities and disadvantages of adopting ML algorithms in the diagnosis of ASD.

Diagnosis of autism is associated with time, cost, and effort. Thus diagnosing a person at an early stage is recommended to reduce the cost and to help the autistic person by prescribing proper medication from early stage. Therefore an effective and time-efficient system is required to predict autism traits accurately.

The objective of this chapter is to explore the existing tree-based ML techniques and propose a new tree-based ML method to predict autism traits of a person at any age. In order to attain this research objective, at the beginning, a good number of previous research works were studied that focus on predicting early autism traits using ML techniques. The literature survey showed that large and precise datasets are required for screening ASD that consist of behavioral traits of previous autistic and nonautistic patients. In this research, AQ-10 dataset [9] was collected and preprocessed for further analysis. A set of real data was collected from 250 ASD and non-ASD cases to further check the reliability of the proposed algorithms. To explore the tree-based prediction models, at first decision tree algorithms Classification and Regression Tree (CART) and Iterative Dichotomiser 3 (ID3) were implemented and tested against the datasets (both AQ-10 and collected real dataset). Later on, as the CART algorithm performed better compared to ID3, random forest classifier based on CART algorithm was implemented by generating array of decision trees. Finally, to improve the prediction performance, a new tree-based approach was proposed that combines ID3 and CART trees in a merged random forest classifier. All the three prediction algorithms were then evaluated on both AQ-10 and collected real datasets. It is worth mentioning here that the previous version of this work is published in Ref. [10]. In this chapter the contribution is extended by doing an in-depth analysis of the tree-based algorithms and discussing how they can be used to model ASD diagnosis tools.

The organization of the remaining sections of this chapter is as follows. Section 2 provides basic idea about tree-based ML methods (specifically decision tree and random forest) and sheds light on the related works done in this field. The overall research methodology has been discussed in Section 3. Sections 4 and 5 consecutively delineate the steps of data analysis and algorithm development. In Section 6 the proposed algorithms have been evaluated on the basis of some well-known metrics that include sensitivity, specificity, and accuracy. Finally, Section 7 concludes the chapter with a brief discussion and possible future implications.

2 Theoretical background and related works

2.1 Machine learning

ML is a technique where the algorithm learns its behavioral traits from the provided input data (training data) instead of being explicitly programmed, and later it classifies new data (testing data) based on its learning. ML algorithms can be mainly divided into two categories: supervised and unsupervised. In supervised learning the input data are labeled into predefined classes, while in unsupervised learning, neither

the classification is predefined nor the input data are labeled. Both regression and classification algorithms come within the scope of supervised ML, and unsupervised ML represents different types of clustering algorithms. In this research work, tree-based classifier algorithms (CART, ID3) have been used which are basically supervised ML techniques.

2.1.1 *Decision tree*

A decision tree is a classifier supervised ML algorithm that uses a tree-like model where each internal node represents a feature of the input dataset and each leaf node represents a class label. The sequence of the features is selected using some statistical calculations. Some common decision tree algorithms are ID3, C4.5 (extension of ID3), CART, CHAID (chi-squared automatic interaction detector), MARS (multivariate adaptive regression splines), etc. In this research work, two well-known decision tree algorithms ID3 and CART have been employed.

2.1.2 *Random forest*

In random forest classifier, multiple decision trees are generated by selecting different combinations and sequences of the attribute nodes randomly, and together they form one single forest. In this method, each decision tree model predicts a class for the input data and the class with the maximum number of votes is selected finally. A combination of multiple decision trees is more likely to reduce overfitting compared to a single decision tree model and thus improves the overall predictive model.

2.2 Overview of the related studies

Several studies have been carried out for predicting autism traits in an individual using different ML techniques. This section briefly introduces some of these studies.

Alternating decision tree (ADTree) was used by Wall et al. [11] to reduce the screening time and to make the detection of ASD traits even faster. Using Autism Diagnostic Interview and Revised (ADI-R) method, they achieved 99.9% accuracy with data of 891 individuals. However, the test data were limited only within the age of 5–17 and thus is not considered as a generalized method of screening ASD. In another work, Bone et al. [12] adopted ML to achieve similar purpose and obtained 89.2% sensitivity and 59% specificity using SVM. Their research was conducted on the basis of 462 individuals with non-ASD traits and 1264 individuals with ASD traits. But, as the age range used for their research was wide (4–55 years), their work did not particularly focus on how the proposed system works to predict ASD among individuals of different age groups like child, adolescent, and adult.

Allison et al. [13] proposed a set of instruments as "Red Flags" for producing shorter version of dataset containing 1000 cases and 3000 controls for screening ASD with a high accuracy. In a recent study, Selvaraj et al. [14] tried to improve the performance of random tree classifier algorithm in ASD prediction (specifically for toddlers) by using effective feature selection algorithms.

Thabtah [15] compared the existing ML-based algorithms for the prediction of autism traits and emphasized on the adaptation of the new DSM-V (Diagnostic and Statistical Manual of Mental Disorders) module for autism screening tools rather than DSM-IV. Similarly, van den Bekerom [16] researched on determining cooccurring conditions with ASD. From the findings, some of these conditions are as follows: developmental delay, obesity, and less physical activity. In order to determine ASD traits in children, they used several ML algorithms, including random forest, SVM, NB, and later compared those results.

An effort was made by Hauck and Kliewer [17] to identify comparatively more effective and important screening questions for both Autism Diagnostic Observation Schedule (ADOS) and ADI-R screening methods. They used SVM technique on ADOS, ADI-R datasets and achieved 85% sensitivity and 80%–90% specificity. They also figured out that when combined together, ADOS and ADI-R screening tests can give better performance. A similar work was done by

Wall et al. [18] where they tried to classify autism with the help of short screening test and validation. They indicated that 8 of the 29 items contained in Module 1 of the ADOS were sufficient to classify autism. They used ADOS Module 1 data from the Autism Genetic Resource Exchange and applied 16 alternative classifiers by performing a series of ML techniques. According to their finding, both ADTree and functional tree performed really well with high values of accuracy specificity, and sensitivity.

Applying deep learning algorithm and neural network, Heinsfeld [19] tried to identify ASD patients from the Autism Brain Imaging Data Exchange (ABIDE I). Using the large brain imaging dataset, he achieved a mean classification accuracy of 70% (sensitivity 74%, specificity 63%), and a range of accuracy of 66%–71%. The SVM classifier achieved mean accuracy of 65% (from 62% to 72%, sensitivity 68%, and specificity 62%), while the random forest classifier achieved a mean accuracy of 63% (sensitivity 69%, specificity 58%). In Ref. [20], Xu et al. used a multilayer artificial neural network along with a sliding window approach to classify children with ASD and typically developing. The classification was done on the basis of short-time spontaneous hemodynamic fluctuations, and a high accuracy of 92.2% could be achieved even with a single optical channel.

Deshpande et al. [21] investigated on effective connectivity among brain areas during intentional causal attribution in ASD and to utilize ML techniques to classify participants based on effective connectivity weights in which they were able to successfully classify participants by diagnosis with 95.9% accuracy.

Liu et al. [22] examined whether face scanning patterns could be potentially useful to identify children with ASD by adopting an ML algorithm for the classification purpose. They have applied SVM to analyze an eye movement dataset from a face recognition task, to classify children with and without ASD and gained a maximum classification sensitivity, AUC, accuracy, and specificity of 93.10%, 89.63%, 88.51%, and 86.21%, respectively.

Kosmicki et al. [23] used ML for evaluating the ADOS to test whether only a subset of behavior is sufficient to decide on traits of ASD and non-ASD among children. They used eight ML algorithms: ADTree, functional tree, LibSVM, logistic model trees, logistic regression, NB, random forest for a large dataset of 4540, and detected ASD risk with 98.27% and 97.66% accuracy for ADOS module 2 and 3, respectively.

In order to identify the issues in conceptual problem formation, methodological implementation, and interpretation, Bone et al. [8] analyzed the previous works of Wall et al. [18] and Kosmicki et al. [23] and used ML approach proposed by them to reproduce the results.

Summary of the previous research works discussed in this section has been presented in

TABLE 9.1 Summary of previous works.

Reference	Technique/algorithm	Prediction results
[11]	Alternating decision tree	Accuracy = 99.9%
[12]	Support vector machine	Specificity = 59%, Sensitivity = 89.2%
[17]	Support vector machine	Specificity = 85%, Sensitivity = 80%–90%
[19]	Neural network Support vector machine random forest	Accuracy = 70%, Specificity = 85%, Sensitivity = 74% Accuracy = 66%, Specificity = 62%, Sensitivity = 68% Accuracy = 63%, Specificity = 58%, Sensitivity = 69%
[22]	Support vector machine	Specificity = 86.21%, Sensitivity = 93.10%
[23]	Eight different machine learning algorithms, i.e., ADTree, functional tree, and LibSVM	Accuracy = 98.27%

Table 9.1. In brief, a wide horizon of research works has been done and still going on to better diagnose the symptoms of ASD using different computational ML algorithms and techniques.

3 Research methodology

The outline of the research methodology is illustrated in Fig. 9.1. First, the AQ-10 dataset was collected from UCI ML repository [9]. Then data were preprocessed and cleaned properly. Later on, best features were selected which contribute most to the prediction variable. Then the data were partitioned into training and test subsets. At first, decision tree algorithm was used to generate the first predictive model using the training set. Using test portion of the data, performance parameters of the model were calculated. Later on applying the same procedures, random forest algorithm was used to create the second predictive model that outperformed decision tree. Finally, a new approach was proposed, merged random forest classifier, that had the best outcomes among all three algorithms. To validate the model, real data was collected from general mass and from an institute of special education for autistic people. Finally, the outcomes of each model were validated using real data.

4 Data collection and analysis

4.1 Data collection

To train an effective predictive model, datasets were collected which consist of AQ-10 screening tool questions as presented in Ref. [9]. The AQ-10 datasets are open-source data and were collected from the UC Irvine ML Repository. Basically, AQ-10 screening tool is used to discern whether or not an individual requires comprehensive autism assessment. It imposes greater

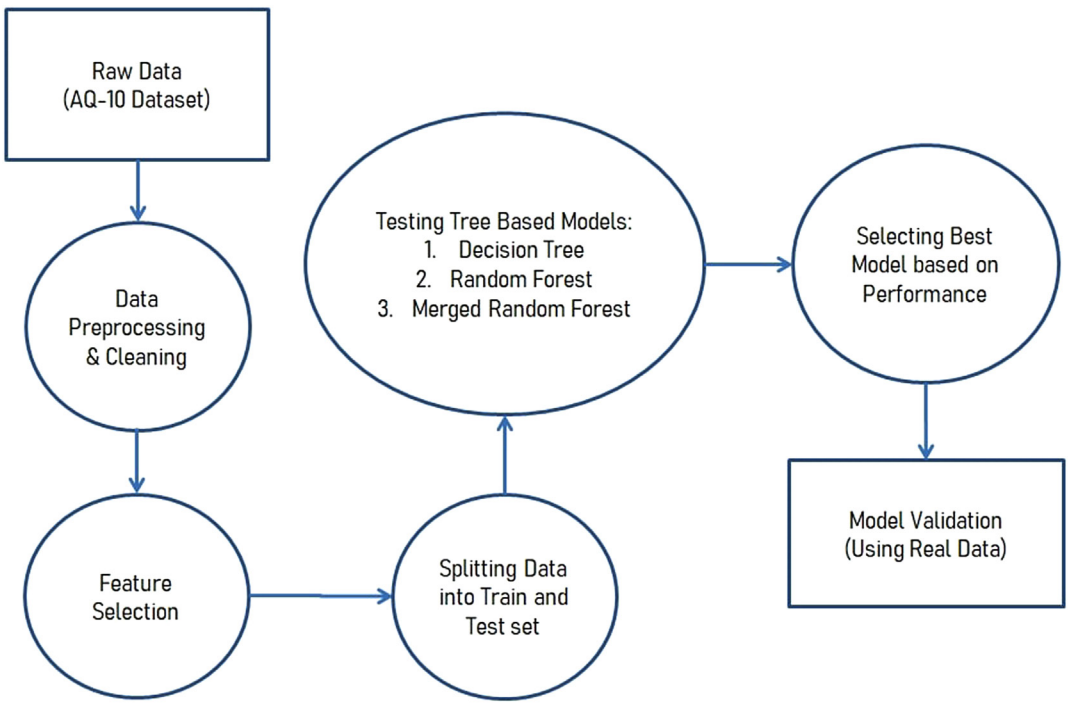

FIGURE 9.1 **Phases of research methodology.**

importance to 10 screening test questions that can easily be completed within a short period of time, and results would guide if user needs extensive autism assessment. There are three different categories of AQ-10: AQ-10 Child (4–11 years), Adolescent (12–16 years), and Adult (18 or more) versions. The datasets consist of data from these three different age brackets.

The AQ-10 questionnaire contains 10 questions that slightly differ from each of the three versions. Fig. 9.2 shows the questionnaires of AQ-10 Adolescent version.

AQ-10 questions are extracted uniformly from five different sections: attention to detail, attention switching, communication, imagination, and social interaction. Users can "slightly" or "definitely" agree or disagree to each of the 10 questions. "Definitely" or "slightly" acquires the same score for all the questions. After answering all the questions, a user gets a score between

NHS
National Institute for Health Research

AQ-10 (Adolescent Version)

Autism Spectrum Quotient (AQ)

A quick referral guide for parents to complete about a teenager aged 12-15 years old with suspected autism who does not have a learning disability.

Please tick one option per question only:		**Definitely Agree**	**Slightly Agree**	**Slightly Disagree**	**Definitely Disagree**
1	S/he notices patterns in things all the time				
2	S/he usually concentrates more on the whole picture, rather than the samll details				
3	In a social group, s/he can easily keep track of several different peaople's conversations				
4	If there is an interruption, s/he can switch back to what s/he was doing very quickly				
5	S/he frequently finds that s/he does't know how to keep a conversatioin going				
6	S/he is good at social chit-chat				
7	When s/he was younger, s/he used to enjoy playing games involving pretending with other children				
8	S/he finds it difficult to imagine what it would be like to be someone else				
9	S/he finds social situations easy				
10	S/he finds it hard to make new friends				

SCORING: Only 1 point can be scored for each question. *Score 1 point for Definitely or Slightly Agree on each items 1, 5, 8 and 10. Score 1 point for Definitely or Slightly Disagree on each of items 2, 3, 4, 6, 7 and 9.* If the individual scores **more than 6 out of 10**, consider referring them for a specialist diagnostic assessment.

FIGURE 9.2 **AQ-10 questionnaires (adolescent) [24].**

TABLE 9.2 Description of features of ASD dataset [9].

Feature	Description
Age	Age in years
Gender	Male/female
Ethnicity	Ethnicity of the participants
Jaundice	Yes/no
Family member with PDD	If a family member has a pervasive development disorder
Who is completing the test	Parent/self/caregiver/medical staff/clinician
Country of residence	Country of the participants
Used the screening app before	If they previously used the screening app
Screening method type	Child/adolescent/adult version
Question 1–10 answer	Answer code 0/1 for each of the ten questions
Result	Final score based on scoring algorithm of the test
Class	Label class if patient has ASD traits or not

0 and 10 [24]. Datasets acquired from UC Irvine ML repository of child, adolescent, and adult versions contain 292, 104, and 704 data samples, respectively. The datasets contain 21 features, including the "Class" column which is our target variable. Detailed description of the features is illustrated in Table 9.2.

4.2 Data preprocessing

The collected datasets had many irrelevant and missing parts in data and those were properly cleaned. The ID column does not affect the probability of having autism traits in an individual; thus the column was dropped. Decision tree algorithm was used to extricate irrelevant attributes from the datasets. The outcome illustrated that removing "relation," "age desc," "used app before," and "age" columns improved overall performance of the prediction models and thus the columns were removed. A summary of the preprocessed datasets is shown in Table 9.3.

TABLE 9.3 Summary of AQ-10 dataset.

Age bracket	Overall preprocessed data instances	Percentage of male versus female	Age (mean) (years)
4–11 Years	248	Male: 70.16 Female: 29.84	6.43
12–16 Years	98	Male: 50 Female: 50	14.13
18 and above	608	Male: 52.7 Female: 47.3	29.63

4.3 Collection of real dataset

In order to validate the research work, true data were collected through survey. Data were collected from two resources: general mass (non-ASD cases) and an institute of special education (ASD cases). The data were collected by on-spot survey, Google forms, and by e-mailing to different academic organizations. Most of the data were collected from on-spot survey with printed forms. These survey questionnaires were prepared using the different attributes from the AQ-10 dataset. The survey was divided into three categories: child version with age group 4–11, adolescent version aging 12–17 years, and adult version with age above 18 years.

Total 62, 31, and 42 instances of real data were collected for child, adolescent, and adult versions. The summary of collected real data is presented in Table 9.4.

5 Developing predictive algorithm

5.1 Decision tree—CART algorithm

Initially, to construct the first predictive model for ASD diagnosis, a tree-based ML algorithm, decision tree—CART, was chosen. ML is an ever evolving sector of artificial intelligence that mimics human intelligence by acquiring knowledge from the surroundings. ML models have been applied successfully in myriad fields ranging from pattern recognition, web

TABLE 9.4 Summary of collected real dataset.

Age bracket	Overall preprocessed data instances	Percentage of male versus female	Percentage of autistic–nonautistic	Age (mean) (years)
4–11 Years	62	Male: 60.2 Female: 30.8	51.61% Autistic 48.39% Nonautistic	7.81
12–16 Years	31	Male: 74.6 Female: 25.4	52.38% Autistic 47.62% Nonautistic	15.21
18 and above	42	Male: 61.9 Female: 30.1	51.61% Autistic 48.39% Nonautistic	23.6

search engines, picture labeling, finance, entertainment, spam detection, medical applications, etc. [25].

Generally, decision tree constructs a tree-based predictive model where each internal node depicts a test on an input attribute, each branch renders result of a test and each leaf node constitutes one of the class labels. The uppermost node of decision tree is called the root node. Decision trees are constructed using algorithms that learn ways to partition data based on divergent conditions. A sample predictive model generated via decision tree is shown in Fig. 9.3.

Initially, the root node of decision tree contains the entire data. Then, data start to split for some feature at each step. Gini impurity and information gain (IG) are used to decide which feature to split on at each stage. Feature with maximum IG is selected to partition data. The splitting procedure continues until the leaf nodes are unmixed or until IG is zero. Gini impurity and IG of decision tree (CART) [26] classifier can be defined as

$$\text{Gini}(\text{data}) = 1 - \sum_{i \in \text{unique\_classes}} P(i)^2 \quad (5.1)$$

$$\text{InfoGain}(\text{data, feature } X) = \text{Gini}(\text{data}) - \sum_{i \in \text{feature}X} \text{AvgGini}(i) \quad (5.2)$$

Implementation of decision tree algorithm can be divided into two phases: (1) constructing the tree (Line 4–14 in Algorithm 1) and (2) classifying test data (Line 16–20 in Algorithm 1). The whole procedure can be described in the following steps:

- At first, best attributes were chosen to generate the predictive model (Line 1) and the two possible class labels were listed (Line 2).
- Next, the train portion of data is passed into "CONSTRUCT_TREE" function (Line 4). Each attribute of the dataset is iterated and the attribute with maximum IG is picked (Line 5–7). If maximum IG is zero, then that indicates that the class label of that fragment of data is unmixed and thus function will return a decision/leaf node (Line 8–10).
- If maximum IG is not equal to zero, then data are partitioned into two fragments (TrueData and FalseData) (Line 11).
- "CONSTRUCT_TREE" function will then run recursively on both segments of the data (TrueData and FalseData) (Line 12–13), and the two edges/branches will create a decision node (Line 14).
- Lastly, after the decision tree model is formed, test portion of data is classified from it. The predictive model classifies each test example by iterating its feature values from the root to some leaf node. The leaf node provides class label for the test case (Line 16–21).

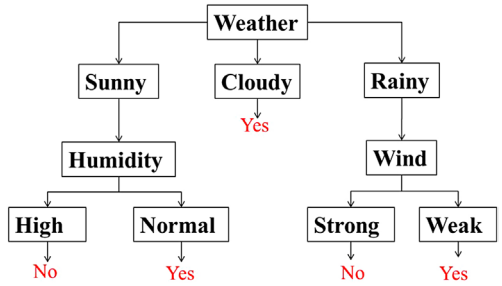

FIGURE 9.3 **A sample decision tree.**

Algorithm 1 Decision tree CART algorithm

```
1:  attributes ← {screening tool questions,
    jaundice, ethnicity, gender, family with PDD,
    etc.}
2:  label_class ← {yes, no}
3:
4:  Function CONSTRUCT_TREE(data)
5:  for each attribute do
6:     calculate max_info_gain
7:  end for
8:  if max_info_gain = 0 then
9:     return leaf_node
10: end if
11: TrueData, FalseData ← Split(data)
12: True_Branch ← CONSTRUCT_
    TREE(TrueData)
13: False_Branch ← CONSTRUCT_
    TREE(FalseData)
14: return Decision_Node(True_Branch, False_
    Branch)
15:
16: Function CLASSIFY_DATA(data, node)
17: if node = leaf_node then
18:    return node.predicions
19: else
20:    Iterate DecisionTree
21: end if
```

5.2 Random forest–CART algorithm

Random forest classifier constructs multiple decision trees using different smaller subsamples of the data each time. It changes the approach of how the standard decision trees are generated. In standard trees, each decision node is created using the best IG among all features.

In a random forest, each individual tree is generated by taking a random sample from the training dataset, which results in different decision trees. Additionally, in a random forest each tree can pick attributes only from a random subset of attributes. This forces even more variation among the trees and imposes more diversity. This tactics turns out to perform very well compared to many other classifiers and also prevents overfitting to a great extent [27]. An outlook of random forest classifier is illustrated in Fig. 9.4.

To improve the performance parameters of the decision tree (CART) model, random forest (CART) algorithm was implemented. The implementation of the algorithm can be divided into two phases: constructing the random forest model (Line 1–11 in Algorithm 2) and classifying the test examples (Line 13–29 in Algorithm 2). Construction of random forest model and its classification process can be described in the following steps:

- First, an array, "array_of_trees," is created to store the decision trees of random forest (Line 4).
- To construct "n" number of decision trees, CONSTRUCT_TREE function is called for "n" times and the produced trees are then appended in "array_of_trees" (Line 5–10).
- Each decision tree is constructed using "p" number of random features. Procedure of constructing decision tree is identical to Algorithm 1 (Line 1–14 of Algorithm 1).
- Lastly, to classify a test example, each of the decision trees from the random forest votes for a class label (Yes/No). If majority of votes

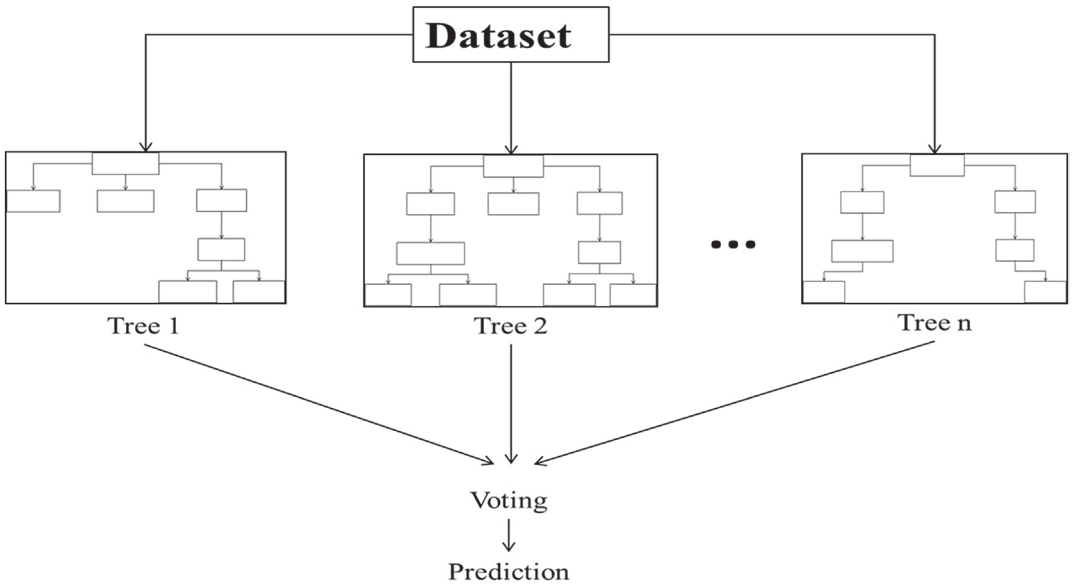

FIGURE 9.4 **Random forest model.**

are "Yes," then the test example is classified as a patient having probable autistic traits or else the patient is classified as having no autistic traits (Line 13–29).

Algorithm 2 Random forest CART algorithm

```
1:  Identical to Line 1 to 14 of Algorithm 1
2:
3:  Function CONSTRUCT_FOREST (data, n,
    train_data_ratio)
4:  array_of_trees ← [ ]
5:  while n ≠ 0 do
6:     train_data ← random(train_data_ratio *
       len(data))
7:     dtree ← CONSTRUCT_TREE(train_data)
8:     dtree_array.append(dtree)
9:     n ← n – 1
10: end while
11: return dtree_array
12:
13: Function CLASSIFY_DATA(data, dtree_
    array[ ], n)
14: j ← 0, vote_y ← 0, vote_n ← 0
15: while j ≠ n do
16:    dtree ← dtree_array(j)
17:    node ← root(dtree)
18:    if node = leaf_node then
19:       if leaf_node.prediction = "Yes" then
20:          vote_y ← vote_y + 1
21:       else if leaf_node.prediction = "No" then
22:          vote_n ← vote_n + 1
23:       end if
24:    else
25:       Iterate_dtree
26:    end if
27:    j ← j + 1
28: end while
29: return vote_y > vote_n
```

5.3 Merged random forest algorithm

In order to improve the performance parameters comparing to decision tree-CART and random forest–CART, an algorithm is proposed that combines the idea of random forest–CART with random forest–ID3. The proposed algorithm can be divided into two phases just as

before: (1) constructing the merged random forest model and (2) classifying test data. In this proposed approach the addition of random ID3 trees made the predictive model more accurate. The entire process is described in the following steps:

- In order to build the predictive model, CONSTRUCT_FOREST function is called and "n" number of ID3 decision trees and "n" number of CART decision trees are generated. Later, the generated trees are stored in an array, "forest_array" (Line 28–38).
- Procedure of constructing ID3 trees (Line 4–14) and CART trees (Line 16–26) is the same as Algorithm 1. Difference between an ID3 tree and a CART tree is that in ID3 decision tree IG is computed using entropy, while in CART decision tree IG is computed using Gini impurity.
- Lastly, to classify a test data, each of the decision trees from the merged random forest votes for a class label (Yes/No). If majority of votes are "Yes," then that instance of test data is classified as a patient having probable autistic traits. On the other hand, if majority of votes are "No," then the patient is classified as having no autistic traits (Line 40–56).

Algorithm 3 Merged random forest algorithm

```
1:  attributes ← {screening tool questions, jaundice, ethnicity, gender, family with PDD etc.}
2:  label_class ← {yes, no}
3:
4:  Function CONSTRUCT_TREE_ID3(data)
5:  for each attribute do
6:     calculate max_info_gain
7:  end for
8:  if max_info_gain = 0 then
9:     return leaf_node
10: end if
11: TrueData, FalseData ← Split(data)
12: True_Branch ← CONSTRUCT_TREE_ID3(TrueData)
13: False_Branch ← CONSTRUCT_TREE_ID3(FalseData)
14: return Decision_Node(True_Branch, False_Branch)
15:
16: Function CONSTRUCT_TREE_CART(data)
17: for each attribute do
18:    calculate max_info_gain
19: end for
20: if max_info_gain = 0 then
21:    return leaf_node
22: end if
23: TrueData, FalseData ← Split(data)
24: True_Branch ← CONSTRUCT_TREE_CART(TrueData)
25: False_Branch ← CONSTRUCT_TREE_CART(FalseData)
26: return Decision_Node(True_Branch, False_Branch)
27:
28: Function CONSTRUCT_FOREST (data, n, train_data_ratio)
29: forest_array ← [ ]
30: while n ≠ 0 do
31:    train_data ← random(train_data_ratio * len(data))
32:    Id3_tree ← CONSTRUCT_TREE_ID3(train_data)
33:    Cart_tree ← CONSTRUCT_TREE_CART(train_data)
34:    forest_array.append(Id3_tree)
35:    forest_array.append(Cart_tree)
36:    n ← n – 1
37: end while
38: return forest_array
39:
40: procedure CLASSIFY_DATA(data, forest_array[ ], n)
41: j ← 0, vote_y ← 0, vote_n ← 0
42: while j ≠ n do
43:    ftree ← forest_array(j)
```

```
44:    node ← root(ftree)
45:    if node = leaf_node then
46:       if leaf_node.prediction = "Yes" then
47:          vote_y ← vote_y + 1
48:       else if leaf_node.prediction = "No" then
49:          vote_n ← vote_n + 1
50:       end if
51:    else
52:       Iterate_ftree
53:       end if
54: j ← j + 1
55: end while
56: return vote_y > vote_n
```

6 Evaluating predictive models

The prediction models would recommend a nondiagnosed person from two possible classes:

"Yes" (Person is likely to have autistic traits), and
"No" (Person does not have autistic traits)

To assess the predictive models, performance parameters, namely, specificity, sensitivity, accuracy, false positive rate, and precision, were calculated. Accuracy of the system is defined by how correctly the machine predicts the condition of the users. Sensitivity calculates the fraction of true positives that are rightly identified as such. Specificity computes the fraction of true negatives that are rightly identified as such. Precision refers to what fraction of positive identifications is actually right. Each of these parameters was calculated for two different datasets (AQ-10 and real dataset). The metrics used to measure the performance can be defined as follows:

$$\text{Specificity} = \frac{TN}{TN + FP}$$

$$\text{Sensivity} = \frac{TP}{FN + TP}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{False-positive rate} = \frac{FP}{TN + FP}$$

$$\text{Accuracy} = \frac{TN + TP}{TN + TP + FN + FP}$$

6.1 Evaluation using AQ-10 datasets

6.1.1 Results of Algorithm 1 (decision tree-CART)

AQ-10 datasets were split into train and test set to evaluate performance parameters of the implemented algorithms. To compute the performance parameters, leave-one-out technique was followed. In leave-one-out technique, each instance of whole dataset is split once. While predicting an instance of data, entire dataset except that instance will be used to train the predictive model. Calculated performance parameters for different age groups are shown in Table 9.5.

The results showed that the decision tree algorithm works better for children (89.92% accuracy) and adult (88.32% accuracy), while for the adolescent it showed only 73.47% accuracy with a false-positive rate of 30.55. The reason behind poor performance on adolescent data is

TABLE 9.5 Results of Algorithm 1 using AQ-10 dataset.

Age group	Accuracy	Specificity	Sensitivity	Precision	False-positive rate
Child	89.92	89.34	90.47	89.76	10.66
Adolescent	73.47	69.44	75.81	81.03	30.55
Adult	88.32	89.95	84.44	77.94	10.04

that the AQ-10 adolescent dataset contains only 98 instances, which may not be sufficient to train the model.

6.1.2 Result of Algorithm 2 (random forest–CART)

Performance parameters of decision tree (CART) were improved by implementing random forest (CART) algorithm on AQ-10 dataset. The leave-one-out technique was used to measure the performance parameters as well. Calculated performance parameters for different age groups are shown in Table 9.6.

The random forest–CART algorithm showed better results in the case of adults with an accuracy of 96.91% and a very low false-positive rate (4.07). It displayed almost a similar performance for child (91.70% accuracy) and adolescent (92.73% accuracy) dataset.

6.1.3 Result of Algorithm 3 (merged random forest classifier)

To further improve the random forest (CART) classifier accuracy, it was then merged with random forest (ID3) classifier and performance parameters indicated that adding ID3 trees to the random forest improves overall performance of the algorithm. Calculated performance parameters are shown in Table 9.7.

Merged random forest algorithm showed promising results for each of the three datasets. It was able to increase the accuracy for adult version of the predictive model up to 97.10%, while accuracy also improved marginally for child (92.26%) and adolescent (93.78%) versions as well. Comparative analysis of the three algorithms for child, adult, and adolescent versions on AQ-10 dataset is displayed in Figs. 9.5–9.7.

6.2 Result analysis using real data

Almost 250 data instances were collected for child, adolescent, and adult versions from an institute of special education. AQ-10 datasets were used to train the prediction model and then real data was tested with the model. Performance parameters calculated on the implemented algorithms are demonstrated in the following subsections.

6.2.1 Result analysis of Algorithm 1 (decision tree-CART)

To validate the study the prediction models were tested for collected real dataset. At first,

TABLE 9.6 Results of Algorithm 2 using AQ-10 dataset.

Age group	Accuracy	Specificity	Sensitivity	Precision	False-positive rate
Child	91.70	88.18	95.72	87.73	12.82
Adolescent	92.73	82.95	98.4	89.85	18.05
Adult	96.91	96.92	96.87	90.07	4.07

TABLE 9.7 Results of Algorithm 3 using AQ-10 dataset.

Age group	Accuracy	Specificity	Sensitivity	Precision	False-positive rate
Child	92.26	88.52	96.52	88.09	12.4
Adolescent	93.78	84.60	98.60	90.82	16.40
Adult	97.10	97.11	97.07	90.54	3.88

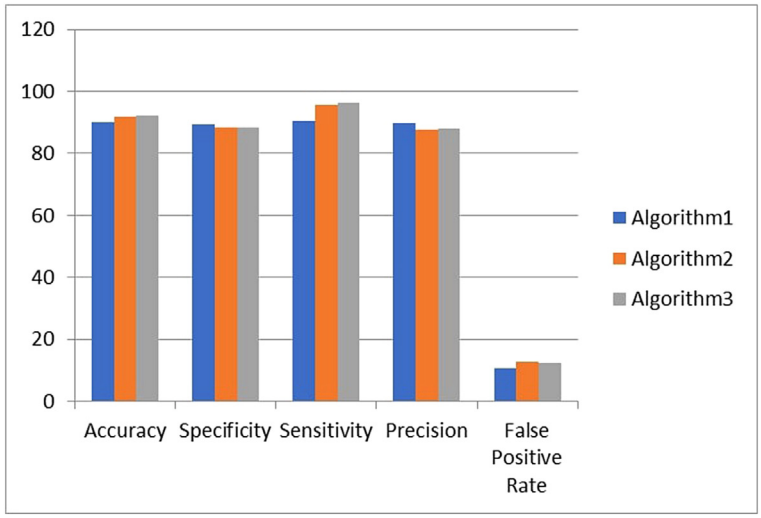

FIGURE 9.5 **Comparison of performance parameters on AQ-10 dataset of child.**

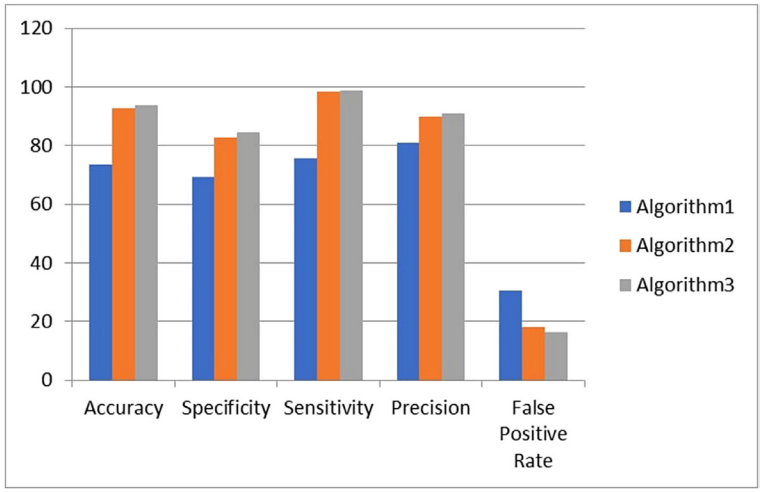

FIGURE 9.6 **Comparison of performance parameters on AQ-10 dataset of adolescent.**

the entire AQ-10 dataset was used to train the model applying decision tree algorithm. Then the model's performance parameters were calculated using collected real data. Calculated performance parameters of decision tree classifier are showed in Table 9.8.

The results from the table illustrate that decision tree classifier works fairly well on adult dataset showing 83.10% accuracy, whereas it underperforms in child (75.04% accuracy) and adolescent (75.89% accuracy) data. The training dataset of AQ-10 adult version contains 608 instances, whereas child and adolescent version contains only 248 and 98 instances, respectively. Due to less instances, the training models of child and adolescent version might be overfitted

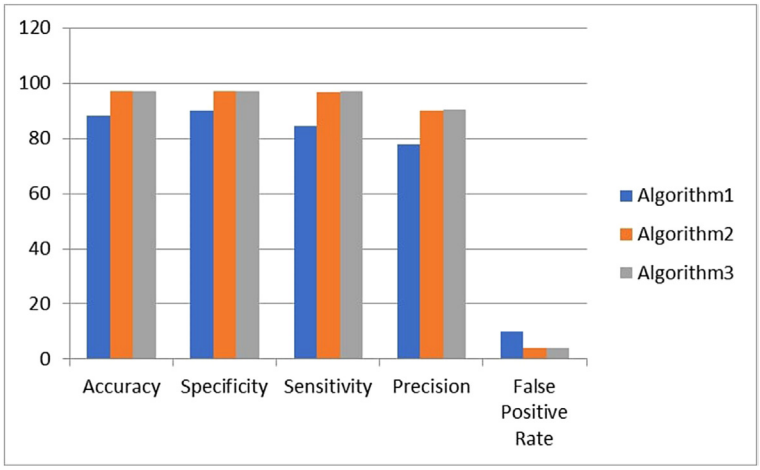

FIGURE 9.7 **Comparison of performance parameters on AQ-10 dataset of adult.**

TABLE 9.8 Results of Algorithm 1 using real data.

Age group	Accuracy	Specificity	Sensitivity	Precision	False-positive rate
Child	75.04	72.57	78.4	71.03	30.40
Adolescent	75.89	69.6	80.7	71.82	33.20
Adult	83.10	81.11	80.07	82.94	12.87

for training data and thus underperforms in real dataset.

6.2.2 Result analysis of Algorithm 2 (random forest–CART classifier)

Random forest–CART classifier was used to build the second predictive model using entire AQ-10 dataset. Then the performance parameters were calculated for the predictive model on collected real dataset. Calculated performance parameters of random forest–CART classifier are illustrated in Table 9.9.

From the results, it seemed that accuracy for each of the three datasets improved marginally with respect to decision tree as random forest algorithm addresses the issue of overfitting. Also, false-positive rate for child (18.66%) and adolescent (16.40%) versions dropped significantly.

TABLE 9.9 Results of Algorithm 2 using real data.

Age group	Accuracy	Specificity	Sensitivity	Precision	False-positive rate
Child	76.92	74.3	80.4	72.76	18.66
Adolescent	77.47	71.27	82.81	73.03	16.40
Adult	84.32	83.95	81.44	83.96	9.02

TABLE 9.10 Results of Algorithm 3 using real data.

Age group	Accuracy	Specificity	Sensitivity	Precision	False-positive rate
Child	77.26	75.34	81.52	73.09	14.48
Adolescent	79.78	71.6	82.6	73.82	13.40
Adult	85.10	84.11	82.07	84.54	6.88

6.2.3 Result analysis of Algorithm 3 (merged random forest classifier)

Lastly, using AQ-10 dataset and the proposed merged random forest classifier, the final predictive model was tested for real dataset. The performance parameters for merged random forest classifier are showed in Table 9.10.

The results illustrate that merged random forest classifier outperformed the previous two algorithms. False-positive rate for all the three datasets decreased significantly, and accuracy and other parameters also had a marginal increase. Comparative analysis of the three algorithms for child, adult, and adolescent versions on real dataset is displayed in Figs. 9.8–9.10.

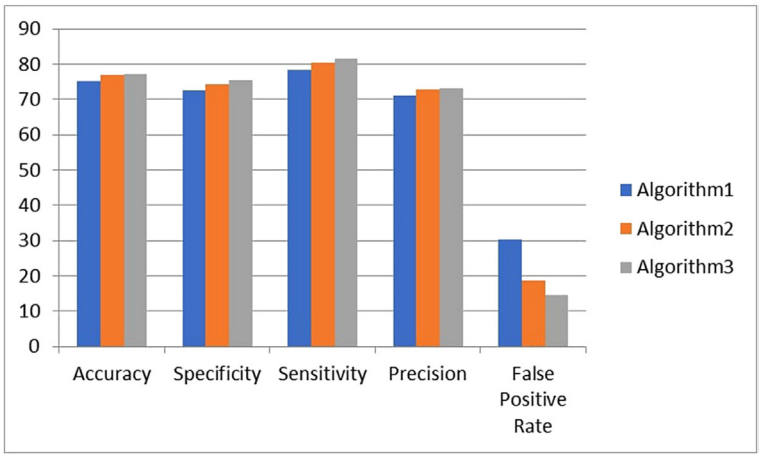

FIGURE 9.8 **Comparison of performance parameters on collected real data of child.**

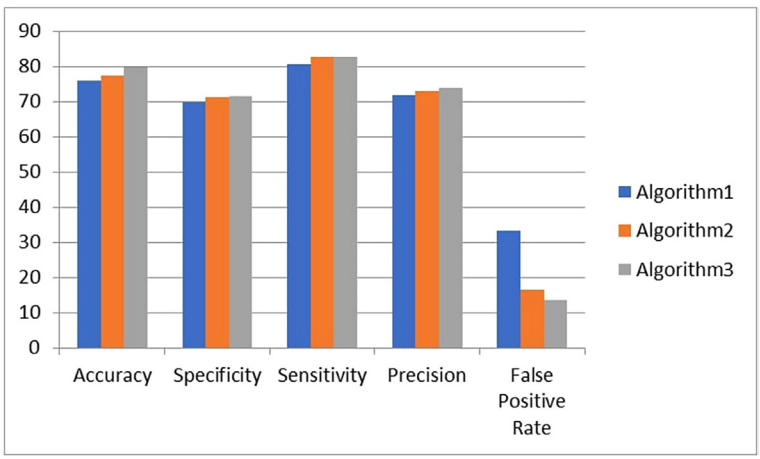

FIGURE 9.9 **Comparison of performance parameters on collected real data of adolescent.**

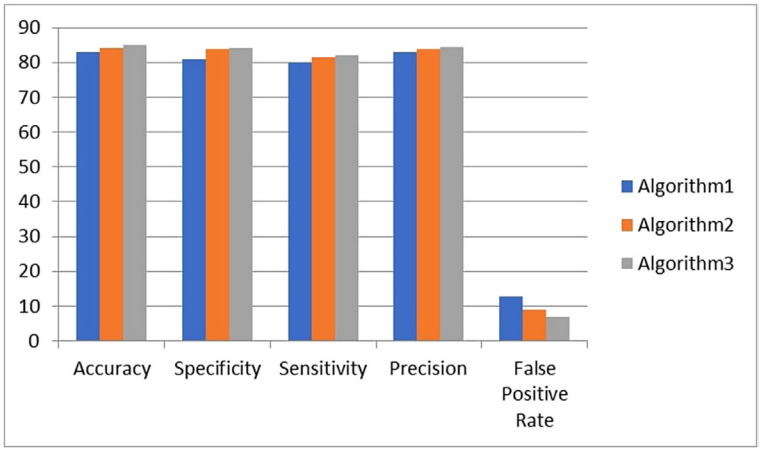

FIGURE 9.10 Comparison of performance parameters on collected real data of adult.

6.3 Comparison of performance on AQ-10 and real dataset

All of the three tree models showed better performance on AQ-10 dataset but performed comparatively poor on collected real dataset. For instance, merged random forest classifier showed an average accuracy of 94.38% on test portion of the three AQ-10 datasets, whereas on real datasets it showed an average accuracy of 80.71%. Additionally, false positive rate of merged random forest on AQ-10 datasets were 10.89% on average, whereas on real dataset it was slightly higher (11.59%). Comparison of the performance gap between the two datasets for merged random forest classifier is illustrated in Fig. 9.11. It can be summarized that the performance parameters calculated for

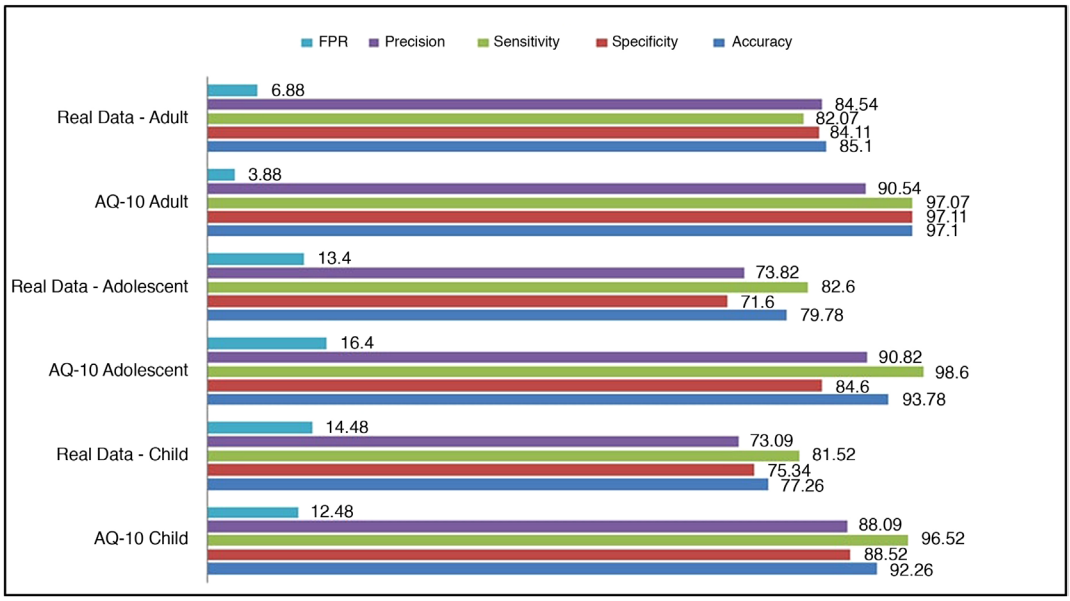

FIGURE 9.11 Comparison of performance parameters between AQ-10 and real dataset (for merged random forest algorithm).

collected real datasets are comparatively poor than AQ-10 datasets. As the real dataset was collected through survey, respondents may not be enough sincere to provide accurate information in many cases and therefore there is a variation of performance parameters between two different datasets.

7 Discussion and conclusion

In this study, AQ-10 dataset was used to develop a predictive model for classifying autism traits in individuals of different age groups by using less number of attributes. In this regard, three different tree-based predictive models were implemented: decision tree, random forest, and merged random forest. The evaluation study showed that the merged random forest classifier provides better results with more than 92% accuracy on the test data for child, adolescent, and adults. The performance parameters were comparatively inferior on collected real dataset as a big portion of those data were collected using online survey methods and thus had some erroneous entries. Overall, the system uses fewer questionnaires and is adequate to provide effective outcomes to find autism traits in an individual.

ASD diagnosis is quite a long process and it is often deferred because of the difficulty for detecting it in children and adolescents. Earlier detection of autism traits can be really beneficial. It would assign an individual for full-time assessment at quite an early stage and thus will lessen long-term costs associated with delayed diagnosis. Also with the help of ML, the system could provide accurate predictions. Many of the existing approaches of ML-based autism screening methods like [11,12] were not generalized for different age groups, whereas the proposed system can detect ASD for three distinct age categories. Also with fewer questionnaires, the system is efficient and more accurate than various existing models like [12,17,19]. Furthermore, the study also provides a comparative view among three different tree-based ML methods and proposes a new approach: merged random forest. From the comparative analysis of the algorithms, it was concluded that the proposed merged random forest classifier outperformed the other two algorithms.

The work has a few limitations as well. First, major setback in implementing the system was availability of sufficiently large dataset. Training datasets related to ASD diagnosis are very rarely open sourced. Moreover, volume of data is also a prime limitation. Acquiring a large dataset could have generalized the prediction model even more. In future, we plan to collect a larger dataset for attaining better performance parameters that will make the prediction model more generalizable.

Second, collection of test data was very challenging. The target group was not comfortable answering odd questions about their children or close ones. Patients were very shy to cooperate and hesitant to participate in survey. Besides, ASD patients are quite rare in the perspective exposure. The parents or guardians of the ASD-affected individual rarely want to expose the patient to any kind of difficult situation.

Third, the system developed provides some fixed category and preset questionnaires that may not be sufficient at times. There might be features/parameters from other screening tools which would make the model perform better. We plan to collect other ASD screening–related questionnaires to address this limitation.

References

[1] U. Frith, F. Happé, Autism spectrum disorder, Curr. Biol. 15 (19) (2005) R786–R790.

[2] M. Randolph-GIps, Autism: a systems biology disease, 2011 IEEE First International Conference on Healthcare Informatics, Imaging and Systems Biology, IEEE, 2011, pp. 359–366.

[3] M. Elsabbagh, G. Divan, Y.J. Koh, Y.S. Kim, S. Kauchali, C. Marcín, C. Montiel-Nava, V. Patel, C.S. Paula, C. Wang, M.T. Yasamy, E. Fombonne, Global prevalence of autism and other pervasive developmental disorders, Autism Res. 5 (3) (2012) 160–179.

[4] S. Bardhan, G.M.M. Mridha, E. Ahmed, M.A. Ullah, H.U. Ahmed, S. Akhter, et al., Autism Barta—a smart device based automated autism screening tool for Bangladesh, 2016 5th International Conference on Informatics, Electronics and Vision (ICIEV), IEEE, 2016, pp. 602–607.

[5] I. Kononenko, Machine learning for medical diagnosis: history, state of the art and perspective, Artif. Intell. Med. 23 (1) (2001) 89–109.

[6] K.K. Hyde, M.N. Novack, N. LaHaye, C. Parlett-Pelleriti, R. Anden, D.R. Dixon, et al., Applications of supervised machine learning in autism spectrum disorder research: a review, Rev. J. Autism Dev. Disord. 6 (2) (2019) 128–146.

[7] S.H. Lee, M.J. Maenner, C.M. Heilig, A comparison of machine learning algorithms for the surveillance of autism spectrum disorder, PLoS One 14 (9) (2019) e0222907.

[8] D. Bone, M.S. Goodwin, M.P. Black, C.C. Lee, K. Audhkhasi, S. Narayanan, Applying machine learning to facilitate autism diagnostics: pitfalls and promises, J. Autism Dev. Disord. 45 (5) (2015) 1121–1136.

[9] F. Thabtah, (2017)*Autistic Spectrum Disorder Screening Data for Children Data Set*, [Online]; available at: https://archive.ics.uci.edu/ml/datasets/Autistic+Spectrum+Disorder+Screening+Data+for+Children++, (accessed 22.08.18).

[10] K.S. Omar, P. Mondal, N.S. Khan, M.R.K. Rizvi, M.N. Islam, A machine learning approach to predict autism spectrum disorder, 2019 International Conference on Electrical, Computer and Communication Engineering (ECCE), IEEE, 2019, pp. 1–6.

[11] D.P. Wall, R. Dally, R. Luyster, J.Y. Jung, T.F. DeLuca, Use of artificial intelligence to shorten the behavioral diagnosis of autism, PLoS One 7 (8) (2012) e43855.

[12] D. Bone, S.L. Bishop, M.P. Black, M.S. Goodwin, C. Lord, S.S. Narayanan, Use of machine learning to improve autism screening and diagnostic instruments: effectiveness, efficiency, and multi-instrument fusion, J. Child Psychol. Psychiatry 57 (8) (2016) 927–937.

[13] C. Allison, B. Auyeung, S. Baron-Cohen, Toward brief "red flags" for autism screening: the short autism spectrum quotient and the short quantitative checklist in 1,000 cases and 3,000 controls, J. Am. Acad. Child Adolesc. Psychiatry 51 (2) (2012) 202–212.

[14] S. Selvaraj, P. Palanisamy, S. Parveen, Autism spectrum disorder prediction using machine learning algorithms, International Conference on Computational Vision and Bio Inspired Computing, Springer, Cham, 2019, pp. 496–503.

[15] F. Thabtah, Autism spectrum disorder screening: machine learning adaptation and DSM-5 fulfillment, in: Proceedings of the 1st International Conference on Medical and health Informatics 2017, 2017. pp. 1-7.

[16] B. van den Bekerom, Using machine learning for detection of autism spectrum disorder, in: Proc. 20th Student Conf. IT, 2017, pp. 120–123.

[17] F. Hauck, N. Kliewer, Machine learning for autism diagnostics: applying support vector classification, Int'l Conf. Heal. Informatics Med. Syst, 2017.

[18] D.P. Wall, J. Kosmicki, T.F. Deluca, E. Harstad, V.A. Fusaro, Use of machine learning to shorten observation-based screening and diagnosis of autism, Transl. Psychiatry 2 (4) (2012) e100.

[19] A.S. Heinsfeld, A.R. Franco, R.C. Craddock, A. Buchweitz, F. Meneguzzi, Identification of autism spectrum disorder using deep learning and the ABIDE dataset, Neuroimage Clin. 17 (2018) 16–23.

[20] L. Xu, X. Geng, X. He, J. Li, J. Yu, Prediction in autism by deep learning short-time spontaneous hemodynamic fluctuations, Front. Neurosci. 13 (2019) 1120.

[21] G. Deshpande, L. Libero, K.R. Sreenivasan, H. Deshpande, R.K. Kana, Identification of neural connectivity signatures of autism using machine learning, Front. Hum. Neurosci. 7 (2013) 670.

[22] W. Liu, M. Li, L. Yi, Identifying children with autism spectrum disorder based on their face processing abnormality: a machine learning framework, Autism Res. 9 (8) (2016) 888–898.

[23] J.A. Kosmicki, V. Sochat, M. Duda, D.P. Wall, Searching for a minimal set of behaviors for autism detection through feature selection-based machine learning, Transl. Psychiatry 5 (2) (2015) e514.

[24] T. Booth, A.L. Murray, K. McKenzie, R. Kuenssberg, M. O'Donnell, H. Burnett, Brief report: An evaluation of the AQ-10 as a brief screening instrument for ASD in adults, J. Autism Dev. Disord. 43 (12) (2013) 2997–3000.

[25] I. El Naqa, R. Li, M.J. Murphy, Machine Learning in Radiation Oncology: Theory and Applications, Springer, 2015.

[26] G. Williams, Decision trees, Data Mining With Rattle and R, Springer, New York, NY, 2011, pp. 205–244.

[27] A. Liaw, M. Wiener, Classification and regression by randomForest, R News 2 (3) (2002) 18–22.