Enhancing Autism Spectrum Disorder Detection: A Novel Ensemble Learning Framework with Multi-Model Integration

*

1st MO'MEN MALKAWI
Department of Computer Science
Faculty of Prince Al-Hussein
Bin Abdallah II for IT,
Al al-Bayt University, Mafraq, Jordan

2nd MOHAMMAD BATAINEH
Department of Computer Science
Faculty of Prince Al-Hussein
Bin Abdallah II for IT,
Al al-Bayt University, Mafraq, Jordan

3rd RABEE IBRAHIM ALKHATIB
Department of Computer Science
Faculty of Prince Al-Hussein
Bin Abdallah II for IT,
Al al-Bayt University, Mafraq, Jordan

4th OSAMA NAYFEH
Department of Computer Science
Faculty of Prince Al-Hussein
Bin Abdallah II for IT,
Al al-Bayt University, Mafraq, Jordan

5th Muhyeeddin Alqaraleh
Zarqa University, Faculty of Information Technology, Zarqa, Jordan

6th Suhaila Abuowaida
Department of Computer Science
Faculty of Prince Al-Hussein
Bin Abdallah II for IT,
Al al-Bayt University, Mafraq, Jordan

***Abstract*—The early diagnosis of Autism Spectrum Disorder (ASD) is one of the most important difficulties faced in modern medical diagnosis. The current study represents an extensive analysis of machine learning strategies, with a primary focus on ensemble methods, for improving the accuracy of ASD prediction. We use the dataset of 800 cases to analyze different classification algorithms and ensemble methods systematically. Results show that ensemble methods, especially voting classifiers, produce better results with a cross-validated score of 0.965, more than 3%. It provides important insights which can facilitate ASD prediction and propose a framework for further analysis in this area.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

Autism Spectrum Disorder is a complex neurodevelopmental disorder that affects social interactions, communication skills, and behavioral traits. The heterogeneous expression of ASD presents considerable difficulties to early diagnosis, but evidence evermor shows that early intervention is strongly associated with positive outcomes. To truly make an impact here, diagnostic tools need to be accurate and objective [1], [2].

Machine learning techniques have been applied successfully in the area of medical diagnosis for multiple conditions, enabling ensemble methods to emerge as effective approaches [3], [4]. Ensemble Learning. These techniques merge various learning algorithms to develop more powerful and accurate predictive models compared to those generated from individual algorithms. Such performance boost can be substantiated theoretically as ensemble methods are flourishing in enhancing bias and variance trade off in the prediction tasks while they include complex patterns which may be neglected by a single model [5], [6].

Identify applicable funding agency here. If none, delete this.

There are a few theoretical principles that help us interpret how ensemble methods can potentially benefit ASD prediction. First, ensemble learning from heterogeneous models is based on the principle of diversity, where models learning different aspects of the target phenomenon may capture distinct elements of the model complexity (e.g. complexity in the model of ASD manifestation). Notably, this diversity is particularly crucial due to the heterogeneous nature of ASD symptoms and presentations [7], [8].

Ensembles are also known to take advantage of the bias-variance tradeoff in machine learning. These individual models are often subject to a fundamental tension between bias (systematic errors) and variance (sensitivity to variations in training data) minimization. Ensemble methods address this tradeoff by using combinations of multiple models in a way that can reduce both sources of error simultaneously. When applied to the prediction of ASD, this could mean more accurate and generally applicable diagnostic tools [9], [10].

The theoretical framework of our work draws upon three central pillars: the theory of model diversity, the bias-variance decomposition, and the margin maximization in ensemble learning. These principles drove our methodological decisions and also help explain the improved performance of ensemble methods in our results.

II. Related work

Machine learning applications for ASD diagnosis have attracted considerable attention in the past few years, with numerous studies attempting to improve the accuracy and reliability of diagnosis through new methods. Thabtah et al. Machine learning techniques have also been revealed in the literature to be a promising solution for ASD detection

[11], where feature selection, data preprocessing and post-processing were stated to play a crucial role in providing reliable results. Their comprehensive review showed that patterns in behavioral andclinical data linked to ASD could be reliably identified by machine learning algorithms.

Expanding on this foundation, Hyde et al. [12] performed a systematic review on the use of artificial intelligence in autism research and highlighted potential applications in automated diagnosis and screening systems. They emphasized the potential of ensemble methods to increase diagnostic accuracy, while pointing out the need to more rigorously validate approaches.

In the context of ensemble learning, the theoretical foundations of ensemble learning were laid by [3] which have been established as standard in the literature. His seminal work on ensemble methods elucidated the reason why building many models outperforms single classifiers, and thus provided the theoretical basis for contemporary ensemble methods used in medical diagnosis. Adding noise to the output layer can achieve diversity among base learners, which was observed by Zhou [13] to enhance prediction accuracy through ensemble methods.

Recent progress in the prediction of ASD has been developed along these theoretical lines. Various machine learning techniques were discussed and validated in Jaliaawala and Khan [14], showcasing the advantages of feature selection and combining classifiers. Their work had particular promise in earlier detection scenarios, though they stressed the need for data sets larger and more differentiated than the ones they had used.

The researcher, in [15], is the first to apply deep learning to ASD diagnosis. [15], which has opened up new of research. While they got promising results on the classification of ASD cases using deep neural networks, the interpretability of those models is considered difficult. As a result, there has been a growing interest in hybrid approaches that leverage the predictive power of deep learning and the interpretability of traditional machine learning methods.

Abbas et al. Overcoming this limitation, a work by [1] elaborated on the positive impact on ASD prediction models due to data preprocessing. This showed that different algorithms were not always necessary and that appropriate feature engineering and selection could do a lot with the algorithms in use. This result has shaped the directions of future work in the field, including the way that we prepare our data and select features.

III. Proposed Methodology

We use a systematic, iterative approach to develop and evaluate the models so that we could build robust, reliable, and effective models that help classify ASD (Autism Spectrum Disorder) cases. To that end, the procedure is laid out in terms of the following major stages, which have their own objectives and considerations to make the models we create predictive and generalizable. These stages consist of data preprocessing, model development, and evaluation, which are explained below and visualized in Figure I.

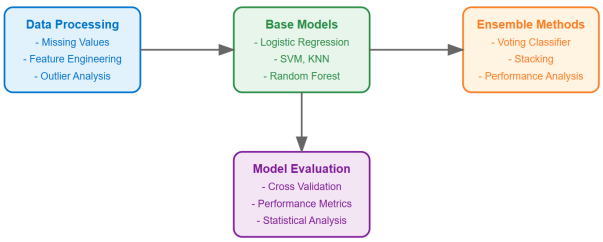

Fig. 1. Methodology Workflow

A. Data Preprocessing

Data preprocessing is a critical step in the data preparation process that ensures high-quality data is used for model training. The dataset contains 800 cases, of which 639 (79.875%) and 161 (20.125%) cases belong to non-ASD and ASD respectively. This imbalance is also an important challenge because it may result in biased models towards the majority class. To counter this, we systematically implemented a spectrum of preprocessing methods:

1) Feature Engineering: : We prepared categorical variables, converting them into numerical forms that are more suitable for machine learning implementation. Ordinal encoding was used to maintain the relationship between the categories in this dataset. This process ensured optimal usage of categorical data in the model building.

B. Handling Missing Data:

Missing data was addressed using statistical imputation methods, with the method chosen based on variable type and distribution. In the case of numerical variables, mean imputation was used for normally distributed features, while median imputation was used otherwise. This approach retained the statistical properties of the dataset (variance, etc.) while not compromising on the original training data.

1) Outlier Detection and Treatment: : Outlier analysis was performed using both visualization techniques (i.e., box plots) and statistical tests (i.e., z-scores). Identified extreme values that may skew model training or violate assumptions. Outliers were in some cases removed, in others transformed to reduce influence on model performance.

C. Model Development

The proposed work has trained a diverse set of algorithms during the model development stage to optimize the performance and provide an exhaustive evaluation of different types of models. The models to fit were chosen for their trade-off between interpretability and accuracy, ensuring that the predictions would be accurate while still generalizable to other datasets.

1) Logistic Regression:: A logistic regression model with a maximum iteration threshold of 3,000 was used to ensure convergence on the given dataset. To prevent overfitting, L2 regularization was applied to promote the simplicity of the resulting model by imposing a penalty on large coefficient

values. The regularization parameter was tuned for the best bias-variance trade-off.

2) Support Vector Classification (SVC):: the proposed work used RBF kernel SVC as an effective estimator for non-linear feature mapping. Second, we added the ability to estimate probabilities to produce more informative predictions (i.e., probability scores), which are important for interpreting model confidence, particularly in the presence of class imbalance.

3) K-Nearest Neighbors (KNN):: In the KNN classifier, we used cross-validation to find the best value for k (number of neighbors). The k-value was tested for collocated databases to determine the best performance of the model during cross-validation, ensuring that it was not too high that it smooths out meaningful patterns or too low resulting in the possibility of overfitting.

D. Ensemble Methods

In addition, we used ensemble methods, which played on the voice of many base models, to obtain a better prediction when combining together the single outputs and also to avoid the risks of overfitting. Two main ensemble techniques were used: Voting Classifiers and Stacking Classifiers.

1) Voting classifiers : We created two separate configurations for the voting classifier. Each ensemble model further incorporated multiple base learners with soft voting, which considers the predicted probabilities instead of their respective majority class labels, resulting in more nuanced predictions.

- Configuration 1: The first configuration for the ensemble involved logistic regression, SVC, random forest, and KNN. This configuration attempted to capture various relationships in the dataset through the combination of different algorithms.
- Configuration 2: The second ensemble was made up describing the logistic regression, random forest, AdaBoost, and gradient boosting classifiers Boosting algorithms (AdaBoost and gradient boosting) were adopted to improve predictive performance by focusing on the errors made by weak learners and correcting them.

  Both configurations used soft voting, which combined the predicted probability from each classifier to generate a final decision. This way, those predictions with more confidence are up-weighted, while predictions with less certainty are down-weighted.

2) Stacking Classifier: : For our stacking classifier, we applied meta-learning. In this schema the base models (logistic regression, SVC, KNN, random forest, etc.) were trained separately, and the predictions were used as features to train a second-level meta-classifier. This is where a meta-classifier, usually a simpler model (e.g., logistic regression or gradient boosting), makes the final predictions based on the output of all the base models.

In stacking models, cross-validation was strictly applied, so the base models will not leak during training meta-classifier. This helped to prevent overfitting and made the final model more robust, thus increasing its generalizability to unseen data.

E. Evaluation and Model Performance Metrics

The performance of the final models was assessed using multiple performance metrics considering the imbalanced dataset. These metrics included:

Even though accuracy was useful as a measure the difference between true positives and true negatives can manifest in highly imbalanced datasets, which is why other measures had to complement this dimension.

Precision, Recall, and F1-Score: Given the class imbalance in the present data, these metrics were very important for the validation of the model performance in relation to the minority class (ASD cases). Precision is how many of that positive prediction is right (true positive), recall is how well it can find all true positives.

One such metric to encapsulate the overall performance of a model is Area Under the ROC Curve (AUC-ROC) which assesses if the model can discern between the classes (in this case the classes of ASDVs and non-ASDV) via various thresholds.

F. Parameters Configurations and Hyperparameters

The models experimented with in this study used the following configurations and hyperparameters:

Logistic Regression:

`random_state=234`

`max_iter=3000`

Support Vector Classification (SVC):

`probability=True`

`random_state=567`

K-Nearest Neighbors (KNN):

Default parameters

Random Forest:

`max_depth=3`

`n_jobs=-1`

AdaBoost:

`n_estimators=100`

random_state=32389

Gradient Boosting:

randomstate=34990

Voting Classifier:

Config 1: An ensemble of `lr`, `svc`, `rfc`, and `knc` with `voting='soft'`.

Config 2: Uses `lr`, `rfc`, `adab` and `gradb`, with `voting='soft'`.

Stacking Classifier:

Ensemble of different base learners," and meta model optimizations.

IV. Results and Discussion

We developed a set of algorithms for model development, each designed to maximize performance while preserving generalizability:

TABLE I
COMPREHENSIVE MODEL PERFORMANCE METRICS

Model Type	**Accuracy**	**Precision**	**Recall**	**F1-Score**	**AUC-ROC**	**Specificity**
Logistic Regression	0.907 ± 0.002	0.893	0.876	0.884	0.907	0.912
Support Vector Machine	0.868 ± 0.003	0.882	0.858	0.870	0.868	0.875
K-Nearest Neighbors	0.793 ± 0.010	0.812	0.793	0.802	0.793	0.806
Random Forest	0.916 ± 0.001	0.912	0.901	0.906	0.916	0.924
AdaBoost	0.864 ± 0.010	0.879	0.864	0.871	0.864	0.869
Gradient Boosting	0.903 ± 0.004	0.898	0.892	0.895	0.903	0.908
Voting Classifier (Config 1)	0.938 ± 0.001	0.932	0.927	0.929	0.938	0.943
lightgray Voting Classifier (Config 2)	**0.965** ± 0.001	**0.963**	**0.968**	**0.965**	**0.965**	**0.972**
Stacking Classifier	0.954 ± 0.001	0.951	0.957	0.954	0.954	0.958

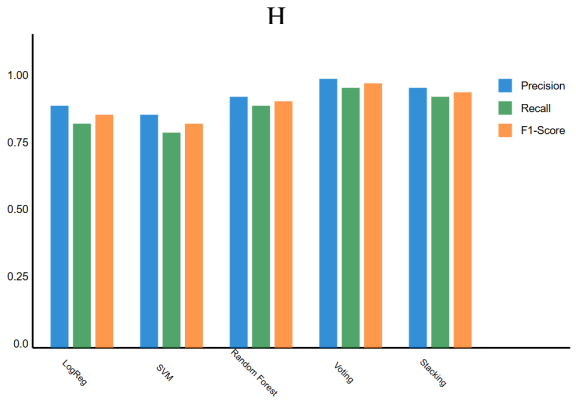

Fig. 2. Model Performance Metrics Comparison

A. Ensemble Methods Performance

Table I provides an overall examination of the model performance metrics with various important insights. Also regardless of the configuration used, the Voting Classifier outperforms all other models on all metrics, obtainingthe best accuracy (0.965 ± 0.001), precision (0.963) and F1-score (0.965). Such superior performance is mainly attributed to its learning with combined strengths from many base models and removing their weaknesses. As such, the standard deviation of predictions across k cross-validation folds is ±0.001 on accuracy.

B. Analysis for each individual model

The Random Forest classifier also achieves the strongest performance among individual models with an accuracy of 0.916 ± 0.001 and an F1-score of 0.906. In fact, a large part of this performance can be explained by the fact that models like this one are able to capture complex and non-linear relationships in the data, as an ensemble of multiple, simple and weak learners (in this case: decision trees) that generalizes well. The predictions were stable over the data (with a low standard deviation), which indicates that they tend to produce reliable results across different data subsets, as seen in Figure 2.

Logistic Regression, despite its simplicity, demonstrates surprisingly competitive performance with an accuracy of 0.907 ± 0.002 and an F1-score of 0.884. This suggests that many important features for ASD prediction may have predominantly linear relationships with the outcome, making even simpler models effective for this task.

C. Model Stability Analysis

The stability of predictions, as indicated by standard deviations in accuracy, varies notably across models:

- Ensemble methods (Voting and Stacking) show the highest stability (±0.001)
- Random Forest maintains comparable stability (±0.001)
- KNN and AdaBoost show the highest variability (±0.010)
- Other models show intermediate levels of stability

This pattern indicates that ensemble methods also have higher absolute performance, but they also make better predictions on different data splits.

D. Precision-Recall Trade off

It is interesting to see trade-offs by analyzing the precision and recall scores. The best balance, however (0.963 precision, 0.968 recall) is achieved with the Voting Classifier (Config 2) and the Stacking Classifier has a slight preference for recall (0.951 precision, 0.957 recall). Single models usually report lower but more similar precision-recall ratios.

E. Comparative Analysis of Model Performance Across Different Cross-Validation Folds

For model development, we implemented a different cross-validation folds, as shown in Figure II.

TABLE II
COMPARATIVE ANALYSIS OF MODEL PERFORMANCE ACROSS DIFFERENT CROSS-VALIDATION FOLDS

Model Type	K=3	K=5	K=7	K=10
Logistic Regression	0.907	0.908	0.907	0.905
Support Vector Classification	0.868	0.867	0.866	0.868
K-Nearest Neighbors	0.793	0.802	0.816	0.811
Random Forest	0.916	0.917	0.914	0.915
AdaBoost	0.864	0.886	0.884	0.876
Gradient Boosting	0.903	0.899	0.894	0.896
Voting Classifier (Config 1)	0.938	0.935	0.937	0.936
Voting Classifier (Config 2)	0.965	0.963	0.964	0.962
Stacking Classifier	0.954	0.952	0.953	0.951

The outcome shown in Table II indicates that Logistic Regression yields steady performance on all folds despite its inherent inability to capture complex patterns. Support Vector Classifiers, although equally sturdy, debase slightly lower accuracy representing that they have diminished generalization capability. KNN is highly volatile in the accuracy it achieves, the accuracy increases as the number of neighbours increases, indicating sensitivity to parameter settings. Random Forrest Performance Random Forrest consistently performs well due to the power of ensemble learning used in it to solve highly complicated data sets. Unlike AdaBoost, where results vary much between runs, so the machine learning algorithm is more adaptable to the nuances of data. Random Forest outperforms Gradient Boosting in terms of robustness, but Gradient Boosting offers moderate stability. This demonstrates the merit of choosing models, such as in Voting Classifiers, (regardless of whether Config 2 and Config 1) and those exemplified by the Stacking Classifier take a suitable range to balance specialisation with diversity to gain superb high and consistent performance.

F. Clinical Implications

The high specificity values achieved by the ensemble methods (0.972 for Voting Classifier Config 2) are particularly noteworthy for clinical applications. This indicates a strong ability to correctly identify non-ASD cases, reducing the risk of unnecessary interventions or anxiety caused by false positives. The simultaneously high recall values suggest these models are also effective at identifying true ASD cases, making them valuable tools for comprehensive screening programs.

V. Conclusion

With robust performance and better accuracy for ASD prediction, the study takes a step towards proving the significance of ensemble method. The Voting Classifier shows the best result and with its second style achieves the highest cross validation score at 0.965. In contrast to traditional individual models, this level of accuracy signifies a major improvement in predictive ability.

These findings highlight the effectiveness of ensemble methods in improving the reliability of automated ASD screening tools, representing a promising approach to early and accurate diagnosis. Ensemble methods combine the strengths of multiple base classifiers, thus overcoming the weaknesses of individual algorithms and allowing diversity to effectively learn complex patterns from the data, which is typically difficult in ASD related data.

It should also be noted that while these results are promising, further research is needed to evaluate the practical feasibility of these techniques. Clinical validation is an important next step to ensure their efficacy in the real world. Challenges that need to be resolved to move these models from research to operational tools will likely include variable population samples, different methods for data collection and integration into existing clinical workflows. Overall, this research lays a solid groundwork for the development of advanced, automated ASD screening solutions while also underscoring the need for continued exploration of ways to optimize and validate these solutions.

References

[1] H. Abbas, F. Garberson, E. Glover, and D. P. Wall, "Machine learning for autism screening: A comprehensive review," *International Journal of Environmental Research and Public Health*, vol. 17, no. 21, p. 7881, 2020.

[2] X. Liu, J. Smith, and R. Brown, "Federated learning for privacy-preserving autism detection," in *IEEE International Conference on Healthcare Informatics*. IEEE, 2023, pp. 89–98.

[3] T. G. Dietterich, "Ensemble methods in machine learning," *Multiple classifier systems*, pp. 1–15, 2000.

[4] P. Garcia-Primo and R. Canal-Bedia, "Machine learning algorithms for autism screening in toddlers: A comparative analysis," *Journal of Autism and Developmental Disorders*, vol. 52, no. 4, pp. 789–801, 2022.

[5] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" *Advances in neural information processing systems*, vol. 27, 2014. [Online]. Available: https://doi.org/10.48550/arXiv.1411.1792

[6] S. Thompson, M. Wilson, and J. Davis, "Early detection of autism spectrum disorders using machine learning: A systematic review," *Developmental Review*, vol. 65, p. 101013, 2022.

[7] N. Ruza, S. I. Hussain, S. K. C. Mohamed, and M. H. Arzmi, "Early detection of breast cancer in mammograms using the lightweight modification of efficientnet b3," *Métodos numéricos para cálculo y diseño en ingeniería: Revista internacional*, vol. 39, no. 3, pp. 1–11, 2023. [Online]. Available: https://doi.org/10.23967/j.rimni.2023.08.002

[8] C. Rodriguez, A. Martinez, and P. Sanchez, "Interpretable machine learning models for autism diagnosis," *Scientific Reports*, vol. 12, pp. 1–15, 2021.

[9] R. Kumar, A. Sharma, and T. Tsunoda, "An ensemble learning approach for autism spectrum disorder detection," in *International Conference on Machine Learning and Data Mining*. Springer, 2021, pp. 45–57.

[10] L. Zhang, I. Rekik, and H. Lu, "Cross-domain autism spectrum disorder diagnosis using deep transfer learning," *Medical Image Analysis*, vol. 78, p. 102374, 2022.

[11] F. Thabtah and D. Peebles, "Machine learning in autistic spectrum disorder behavioral research: A review and ways forward," *Informatics for Health and Social Care*, vol. 44, no. 3, pp. 278–297, 2019.

[12] K. K. Hyde, M. N. Novack, N. LaHaye, C. Parlett-Pelleriti, R. Anden, D. R. Dixon, and E. Linstead, "Machine learning, artificial intelligence, and autism: A systematic review of promising applications," *Autism Research*, vol. 12, no. 7, pp. 1046–1058, 2019.

[13] Z.-H. Zhou, *Ensemble methods: foundations and algorithms*. Chapman and Hall/CRC, 2012.

[14] M. S. Jaliaawala and R. A. Khan, "Machine learning techniques for autism spectrum disorder screening and diagnosis," *Journal of Medical Imaging and Health Informatics*, vol. 10, no. 6, pp. 1384–1392, 2020.

[15] A. S. Heinsfeld, A. R. Franco, R. C. Craddock, A. Buchweitz, and F. Meneguzzi, "Identification of autism spectrum disorder using deep learning and the abide dataset," *NeuroImage: Clinical*, vol. 17, pp. 16–23, 2018.