

Machine Learning Approach to Classification of Childhood Disabilities for Special Education

Mun-Taek Choi¹, Tae-Seong Kim² and Jung-bae Kang³

¹ SKKU Advanced Institute of Nano Technology, Sungkyunkwan University, Republic of Korea

² Dept. of Biomedical Engineering, Kyung Hee University, Republic of Korea

³ Research Planning Team, Korea Disabled People's Development Institute, Republic of Korea

Abstract: *This paper presents an investigation of a machine learning approach to classification of childhood disabilities for special education, an inherently complicated multi-class problem. We applied the supervised learning to train various classification algorithms utilizing an exhaustive grid search technique with cross validation to find optimal hyper-parameters for the classifiers. The analysis results showed that the outperformed classifiers successfully classified childhood disabilities with the F_1 score of 0.82. This study shows highly confident applicability of the machine learning approach to the classification of childhood disabilities in the field of special education.*

Keywords: *Special Education, Childhood Disabilities, Classification, Supervised Learning, Grid Search*

1. Introduction

The number of children with disabilities steadily increased. For example, in the United States, there has been 15.6% increase in the percentage of childhood disabilities between 2001 and 2011 [1]. In Korea, there has been 33.6% increase between 2007 and 2015 [2].

To provide appropriate special education to needed children in a timely manner, classifying or identifying the categories of disability that children may have is crucial [3]. The diagnostic and evaluation procedures for children with disabilities are usually very complex. Therefore, the investigation or development of systematic evaluation methodologies or diagnostic tools is demanded.

Many researches have tried various machine learning approaches to investigate possibility of improving classification of childhood disabilities. Kosmicki et al. employed feature selection with stepwise backward to evaluate Autism Diagnostic Observation Schedule that identify ADS with a subset of behaviours of children [4]. Crippa et al. used supervised-learning method to determine if upper-limb movement could be used to classify low-functioning children with ASD aged 2–4 [5]. They found potential possibility of identifying subset of patients using Support Vector Machine (SVM), but the number of samples were limited by 15 children only.

Mueller et al. adapted SVM with feature selection to see if features of independent Event Related Potentials (ERP) components could be used for the classification of ADHD [6]. Tenev et al. applied SVM to classify ADHD patients from control group [7]. Four SVM classifiers were trained with different data set, taken for four different conditions, and then the output of the classifiers is logically combined.

In this study, we investigated multi-class classification of childhood disabilities utilizing wide spectrum of classification algorithms with optimal parameters on the performance metrics. The remaining paper is organized as follows. Section 2 clarifies technical details of how to obtain classification models with optimal hyper-

parameters to the problem of identification of childhood disabilities. In Section 3, results and discussions are presented on exhaustive comparison of performance measures of various classification models. Section 4 addresses conclusions of this work.

2. Methods

The description of sample data used in this study is presented. A brief introduction to classification algorithms is provided. The grid search method to find optimal hyper-parameters per classifier by comparing performance metrics is explained.

2.1. Sample Data Set

The sample data used in this analysis were obtained from the official evaluation results of 179 children with disabilities between the ages of 2 and 15 by the Special Education Centers in Daegu, Republic of Korea from January to December 2014. There are 5 categories of childhood disabilities in this analysis: Developmental Delay (DD), Autism Spectrum Disorder (ASD), Emotional or Behavioral Disorder (EBD), Intellectual Disability (ID) and Physical Disability (PD). Since each child belongs to only one category for special education, it becomes the problem of multi-class classification.

In the data set, the labels are given by the categories of childhood disabilities, officially determined by the committee in the centers. There are over 136 features derived from various assessment scores including K-ABC, K-ABC-II, K-WISC-IV, KNISE-SAB, SELSI, PRES, U-TAP, K-SIB-R, CBCL 6-18, K-CDI, TRF, C-TRF, CBCL 1.5-5, K-ADHDDS, SCQ, STROOP, KOSECT, REVT and K-ADS. The letter K as a prefix represents the Korean version of the corresponding international standard evaluation kits for special education.

2.2. Model Learning

To design a good classification model or classifier for the given data, it is important to experiment with various classification algorithms and fully understand the performance characteristics of the algorithms. In this research, we picked popular classification algorithms as classifiers covering various types, i.e., Logistic Regression [8], Support Vector Machine [9], Random Forest [10], Gradient Boosting [11] and Nearest Neighbors [12]. In this study, all the algorithms are implemented using Scikit-learn, a powerful machine learning library in Python [13].

To avoid overly generalizing classification performance of classifiers, it is important to measure performance with the different data, never seen during model learning. We partitioned the sample data into two sets of 75% for training and 25% for test, let the classifiers learned from the training set and measured the classification performance using the test set.

A classifier exhibits different performances with different hyper-parameter settings. Finding optimal settings for classifiers becomes the problem of grid search with exhaustive scan of various hyper-parameter combination. Table I shows classifiers for the grid search and ranges of hyper-parameters in this implementation. We defined key names of classifiers and determined ranges of discrete hyper-parameter values per classifiers for the parameter scan.

TABLE I: Values of hyper-parameters per classifier in grid search

Classifier	Parameters
GBC	{'n_estimators': [32, 64, 128], 'learning_rate': [0.6, 1.0, 1.4]}
KNC	{'n_neighbors': [1, 3, 5]}
LR	{'C': [0.01, 1, 100]}
RFC	{'n_estimators': [32, 64, 128]}
SVC	{'kernel': ['linear'], 'C': [0.1, 10, 100]}
	{'kernel': ['rbf'], 'C': [10, 100, 1000], 'gamma': [0.001, 0.01, 0.1]}

* GBC: Gradient Boosting, KNC: K-Nearest Neighbors, LR: Logistic Regression, RFC: Random Forest, SVC: Support Vector Machine

* n_estimators: # of trees in the forest or boosting stages, learning-rate: decrease of the contribution of each tree, n_neighbors: # of neighbors, C: inverse of regularization strength, kernel: kernel type, gamma: kernel coefficient, rbf: Radial Basis Function

During evaluating classifiers with different hyper-parameters, the classifiers should be validated with different data from model learning to avoid generalization error. To avoid further partitioning of the data for the validation, we used 10-fold stratified cross validation.

The quality of classification of a model is represented by performance metrics. The performance metrics of this multiclass classification used in this study include accuracy, precision, recall and F₁ score. Calculation of metric scores on multiclass problem was done by one-vs.-all method, which is the most commonly used strategy [14]. To consider the class imbalance, we calculated the scores with the “weighted” average in which each class’s score is weighted by its size in the sample data. Moreover, the quality of classification is visually represented by confusion matrices in a table layout.

3. Results

The performance scores from the whole spectrum of the grid search on the training set are shown in Table II including hyper-parameters, accuracy, precision, recall and F₁ score. The best performing parameters for each classifier based on the F₁ scores are marked with bold faces in Table II.

TABLE II: Performance scores of the classifiers on the training set

Classifier	Parameters	Accuracy	Precision	Recall	F ₁ Score
GBC	{'n_estimators': 32, 'learning_rate': 0.6}	0.7520	0.7548	0.7520	0.7491
	{'n_estimators': 64, 'learning_rate': 0.6}	0.7280	0.7628	0.7520	0.7599
	{'n_estimators': 128, 'learning_rate': 0.6}	0.7440	0.7557	0.7760	0.7403
	{'n_estimators': 32, 'learning_rate': 1.0}	0.7760	0.7766	0.7840	0.7501
	{'n_estimators': 64, 'learning_rate': 1.0}	0.7920	0.7546	0.7760	0.7609
	{'n_estimators': 128, 'learning_rate': 1.0}	0.7840	0.7846	0.7760	0.7581
	{'n_estimators': 32, 'learning_rate': 1.4}	0.7200	0.7392	0.6800	0.7097
	{'n_estimators': 64, 'learning_rate': 1.4}	0.7120	0.6981	0.7280	0.6712
KNC	{'n_neighbors': 1}	0.6480	0.6824	0.6480	0.6426
	{'n_neighbors': 3}	0.6880	0.7156	0.6880	0.6843
	{'n_neighbors': 5}	0.6800	0.6474	0.6800	0.6445
LR	{'C': 0.01}	0.7360	0.7189	0.7360	0.7040
	{'C': 1}	0.7520	0.7755	0.7520	0.7468
	{'C': 100}	0.6880	0.6945	0.6880	0.6728
RFC	{'n_estimators': 32}	0.7760	0.7531	0.7440	0.7325
	{'n_estimators': 64}	0.7760	0.7401	0.7680	0.7412
	{'n_estimators': 128}	0.7760	0.7283	0.7600	0.7523
SVC	{'kernel': 'linear', 'C': 0.1}	0.6960	0.6815	0.6960	0.6692
	{'kernel': 'linear', 'C': 10}	0.7120	0.7323	0.7120	0.7051
	{'kernel': 'linear', 'C': 100}	0.7120	0.7300	0.7120	0.7033
	{'gamma': 0.001, 'kernel': 'rbf', 'C': 10}	0.7200	0.6521	0.7200	0.6622
	{'gamma': 0.01, 'kernel': 'rbf', 'C': 10}	0.7120	0.7244	0.7120	0.6923
	{'gamma': 0.1, 'kernel': 'rbf', 'C': 10}	0.6560	0.5663	0.6560	0.5893
	{'gamma': 0.001, 'kernel': 'rbf', 'C': 100}	0.7440	0.7578	0.7440	0.7277
	{'gamma': 0.01, 'kernel': 'rbf', 'C': 100}	0.7280	0.7578	0.7280	0.7168
	{'gamma': 0.1, 'kernel': 'rbf', 'C': 100}	0.6480	0.5593	0.6480	0.5822
	{'gamma': 0.001, 'kernel': 'rbf', 'C': 1000}	0.7360	0.7540	0.7360	0.7219
{'gamma': 0.01, 'kernel': 'rbf', 'C': 1000}	0.7040	0.7245	0.7040	0.6837	
{'gamma': 0.1, 'kernel': 'rbf', 'C': 1000}	0.6480	0.5593	0.6480	0.5822	

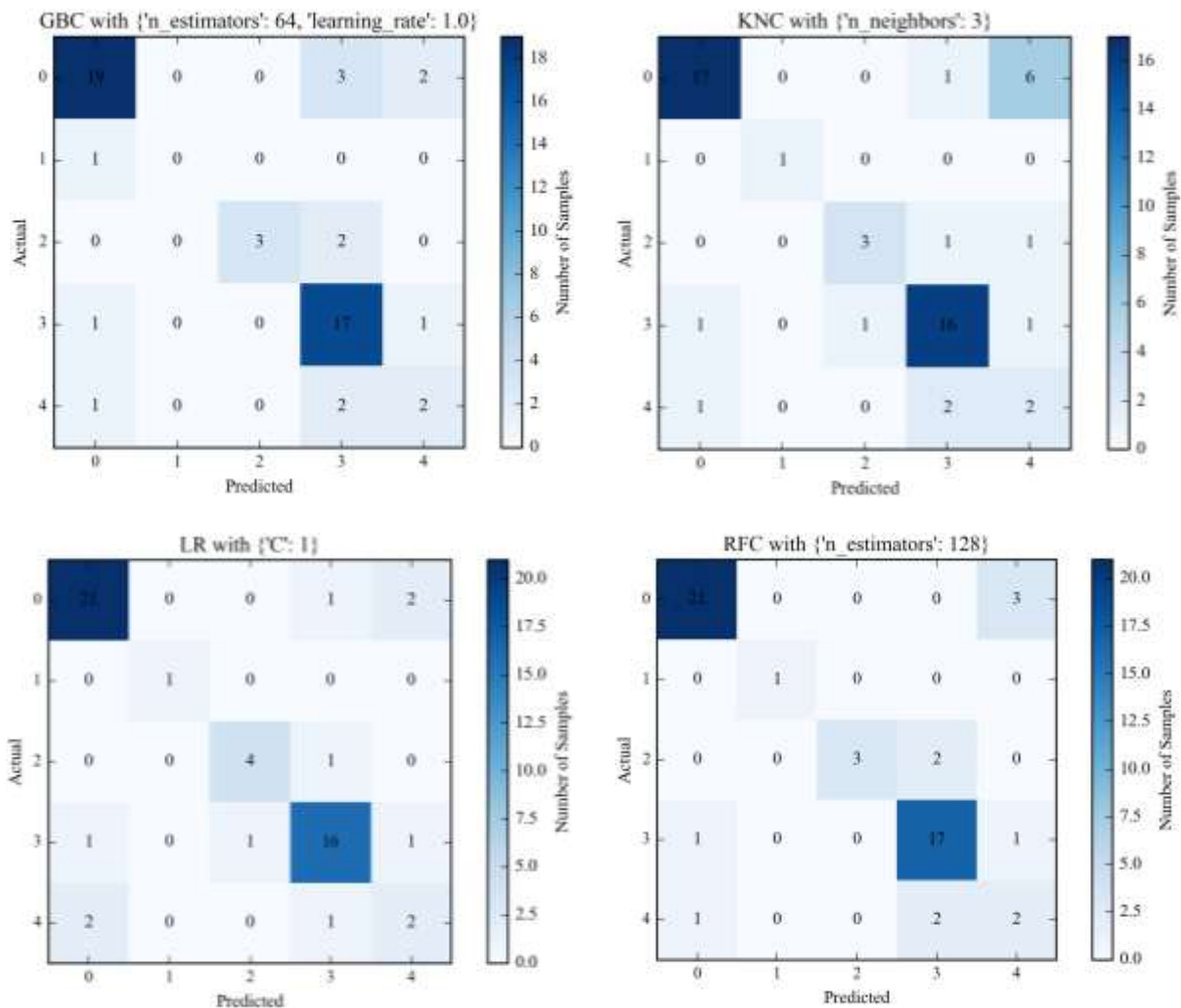
* All the scores are the weighted averages, calculated for each class and weighted by the number of samples in each class.

We picked the best performing parameters for each classifier and evaluated the classifier with the best parameters using the test set that we did not include during the training. Table III shows the evaluation results on the test set, i.e., the performance metrics. For the classification of childhood disability, LR and RFC outperformed other classifiers. LR and RFC achieved the F_1 score of about 0.82, which means the classifiers classify children with disabilities as good as 82% of the best human experts in the field. As is practically meaningful if the confidence rate is over 80% [15], this machine learning approach shows highly confident result on the classification of childhood disabilities in the special education.

TABLE III: Performance scores of the classifiers on the test set

Classifier	Parameters	Accuracy	Precision	Recall	F_1 Score
GBC	{'n_estimators': 64, 'learning_rate': 1.0}	0.7593	0.7627	0.7593	0.7518
KNC	{'n_neighbors': 3}	0.7222	0.7856	0.7222	0.7451
LR	{'C': 1}	0.8148	0.8148	0.8148	0.8148
RFC	{'n_estimators': 128}	0.8148	0.8326	0.8148	0.8179
SVC	{'kernel': 'linear', 'C': 10}	0.7593	0.8078	0.7593	0.7691
	{'kernel': 'rbf', 'C': 100, 'gamma': 0.001}	0.7593	0.7670	0.7593	0.7621

To visualize and confirm the performance of the classifiers, confusion matrices are presented in the Fig. 1.



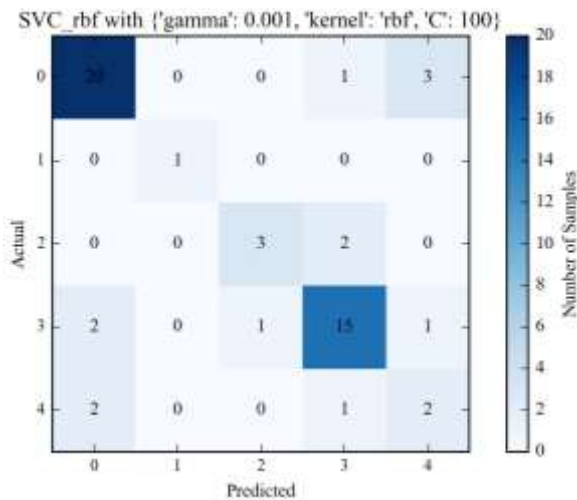


Fig. 1: Confusion matrices of classifiers (axis: 0 - DD, 1 - ASD, 2 - EBD, 3 - ID and 4 - PD).

4. Conclusion

This study seeks a machine learning approach that successfully classify childhood disabilities for special education, inherently a complicated multi-class classification problem. We applied the grid search technique with cross validation to reveal optimal hyper-parameters of various classifiers. The results showed that the classifiers with optimal hyper-parameters could classify the childhood disabilities with the F_1 score of 0.82. This clearly shows highly confident applicability of the machine learning approach to the classification of childhood disabilities for special education. Consequently, properly trained classifiers can be a practical predictor for use in the diagnosis of childhood disability.

5. Acknowledgements

This material is based upon work supported by the Ministry of Trade, Industry & Energy (MOTIE, Korea) under Industrial Technology Innovation Program. No. 10048451, ‘Development of a Diagnostic and Training Robot System for Students with Autism and ADHD, compliant to 95% of Human Diagnostic Results’

6. References

- [1] A. J. Houtrow, K. Larson, L. M. Olson, P. W. Newacheck, and N. Halfon, “Changing Trends of Childhood Disability, 2001–2011,” *PEDIATRICS*, vol. 134, no. 3, pp. 530–538, Sep. 2014.
<https://doi.org/10.1542/peds.2014-0594>
- [2] Special Education Policy Division, Special Education Annual Report, Ministry of Education of Korea, Sep. 2015 (Korean)
- [3] L. Florian, J. Hollenweger, R. J. Simeonsson, K. Wedell, S. Riddell, L. Terzi, and A. Holland, “Cross-Cultural Perspectives on the Classification of Children with Disabilities: Part I. Issues in the Classification of Children with Disabilities,” *Journal of Special Education*, vol. 40, no. 1, pp. 36–45, Mar. 2006.
<https://doi.org/10.1177/00224669060400010401>
- [4] J. A. Kosmicki, V. Sochat, M. Duda, and D. P. Wall, “Searching for a Minimal Set of Behaviors for Autism Detection through Feature Selection-based Machine Learning,” *Translational Psychiatry*, vol. 5, no. 2, p. e514, 2015.
<https://doi.org/10.1038/tp.2015.7>
- [5] A. Crippa, C. Salvatore, P. Perego, S. Forti, M. Nobile, M. Molteni, and I. Castiglioni, “Use of Machine Learning to Identify Children with Autism and Their Motor Abnormalities,” *J Autism Dev Disord*, vol. 45, no. 7, pp. 2146–2156, Jul. 2015.
<https://doi.org/10.1007/s10803-015-2379-8>

- [6] A. Mueller, G. Candrian, J. D. Kropotov, V. A. Ponomarev, and G.-M. Baschera, "Classification of ADHD Patients on the basis of Independent ERP Components using a Machine Learning System," *Nonlinear biomedical physics*, vol. 4, no. 1, p. S1, 2010
<https://doi.org/10.1186/1753-4631-4-S1-S1>
- [7] A. Tenev, S. Markovska-Simoska, L. Kocarev, J. Pop-Jordanov, A. Müller, and G. Candrian, "Machine Learning Approach for Classification of ADHD Adults," *Int J Psychophysiol*, vol. 93, no. 1, pp. 162–166, Jul. 2014.
<https://doi.org/10.1016/j.ijpsycho.2013.01.008>
- [8] H. F. Yu, F. L. Huang, and C. J. Lin, "Dual Coordinate Descent Methods for Logistic Regression and Maximum Entropy Models," *Machine Learning*, vol. 85, no. 1, pp. 41–75, 2011.
<https://doi.org/10.1007/s10994-010-5221-8>
- [9] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, pp. 27–27, 2011
<https://doi.org/10.1145/1961189.1961199>
- [10] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001
<https://doi.org/10.1023/A:1010933404324>.
- [11] J. H. Friedman, "Stochastic gradient boosting," *Computational Statistics & Data Analysis*, vol. 38, no. 4, pp. 367–378, 2002.
[https://doi.org/10.1016/S0167-9473\(01\)00065-2](https://doi.org/10.1016/S0167-9473(01)00065-2)
- [12] T. M. Cover and P. E. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inform. Theory*, vol. 13, no. 1, pp. 21–27, 1967.
<https://doi.org/10.1109/TIT.1967.1053964>
- [13] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [14] J. Milgram, M. Cheriet, and R. Sabourin, "'One Against One' or 'One Against All': Which One is Better for Handwriting Recognition with SVMs?," presented at the Tenth International Workshop on Frontiers in Handwriting Recognition, 2006.
- [15] J. C. Nunnally and I. H. Bernstein, *Psychometric Theory*, 3rd ed. New York: McGraw-Hill, 1994