Prediction of Students' Study Club Interest Using Naive Bayes and Random Forest Algorithms Based on Academic Scores

Arip Muhridan, Andysah Putera Utama Siahaan

Abstract
An inappropriate study club choice may reduce students' learning effectiveness. This study compares the performance of the Naive Bayes and Random Forest algorithms in predicting students' study club interest from academic scores. The data consist of supporting-course scores of 1,079 students from the Computer Systems and Information Systems study programs, with five predictors (supporting-course score, attendance, project score, midterm, and final exam) and four interest classes: Data Science, Web Programming, IoT, and Network Engineering. Gaussian Naive Bayes and Random Forest (100 trees) were evaluated using five-times repeated stratified 5-fold cross-validation and an 80:20 hold-out test. Random Forest outperformed Naive Bayes, with an accuracy of 47.99% and a macro F1-score of 0.395, compared with 40.35% and 0.242, respectively. However, the accuracy of Random Forest is only slightly above the majority-class baseline (44.21%), indicating that academic scores alone are not a strong predictor of interest. The supporting-course score was the most important feature (0.317). Future work should add interest-related variables and address class imbalance.

Keywords: Interest Prediction, Study Club, Naive Bayes, Random Forest, Academic Scores, Educational Data Mining

Arip Muhridan[1]
[1]Magister Teknologi Informasi, Universitas Pembangunan Panca Budi, Indonesia
e-mail: aripmuhridan14@gmail.com[1]

Andysah Putera Utama Siahaan[2]
[2]Magister Teknologi Informasi, Universitas Pembangunan Panca Budi, Indonesia
email: andiesiahaan@gmail.com[2]
2nd International Conference on Islamic Community Studies (ICICS)
Theme: History of Malay Civilization and Islamic Human Capacity and Halal Hub in the Globalization Era
https://proceeding.pancabudi.ac.id/index.php/ICIE/index

Introduction

Study clubs are platforms for developing students' competencies outside regular coursework, focusing on specific areas of expertise such as web development, data science, the Internet of Things (IoT), and network engineering. Choosing the right study club plays an important role in sustaining learning motivation and guiding students toward fields that match their abilities. A choice based only on trends, peer invitations, or perceived difficulty may create a mismatch between interest and academic ability [1].

Universities already hold rich academic data, but this data has not been fully used to support student development decisions. Educational Data Mining (EDM) offers an approach for discovering patterns in academic data to support decision-making [2], [3]. Classification techniques have been widely used to predict student performance and graduation [4], [5], as well as to map specializations or majors, for example using Decision Tree and Naive Bayes [6], [1].

Several previous studies are relevant to this topic. Tamrin [7] developed a prediction model for Information Systems students' specialization based on academic performance using Decision Tree, Random Forest, and XGBoost, and reported Random Forest as the best model after applying SMOTE and hyperparameter tuning. Munawirah and Arisha [1] used Naive Bayes to classify study program preferences among new students based on gender, school of origin, and school major, and suggested adding academic scores and comparing against Random Forest in future work. Mahardika et al. [8] mapped the distribution of students' interests in informatics concentrations using a data-driven approach. In the context of academic achievement, Yusuf et al. [9], Adnyana [10], and Budiyanto et al. [11] compared Naive Bayes and Random Forest, with Random Forest tending to give higher accuracy on student score data.

Nevertheless, studies that specifically compare Naive Bayes and Random Forest for predicting study club interest from supporting-course scores remain limited. This study differs in that it uses assessment component scores (supporting-course score, attendance, project score, midterm, and final exam) from two study programs with four study club classes, and reports model performance honestly against the majority-class baseline. The objectives of this study are (1) to build a study club interest prediction model using Naive Bayes and Random Forest, (2) to compare the performance of the two algorithms, and (3) to identify the most influential academic variables.

Literature Review

2.1 Educational Data Mining and classification

Educational Data Mining is the application of data mining methods to educational data to understand learning processes and support academic decisions [3]. Classification is the process of building a model that maps a set of attributes to predefined classes so that the class of new data can be predicted [12]. In this study, the attributes are academic scores and the class is the student's study club interest.

2.2 Naive Bayes

Naive Bayes is a probabilistic classifier based on Bayes' theorem, assuming that each attribute is conditionally independent given the class [13]. Although this assumption is rarely satisfied, the classifier often performs reasonably well and efficiently [14]. The posterior probability of the k-th class for an attribute vector x is given in (1), where P(C) is the class prior probability. For numerical attributes, Gaussian Naive Bayes assumes a normally distributed likelihood as in (2), where μ and σ^2 are the mean and variance of the i-th attribute in the k-th class, respectively.

$$P(C_k \mid \mathbf{x}) = \frac{P(C_k) \prod_{i=1}^{n} P(x_i \mid C_k)}{P(\mathbf{x})} \tag{1}$$

$$P(x_i \mid C_k) = \frac{1}{\sqrt{2\pi\sigma_{ik}^2}} \exp\left(-\frac{(x_i - \mu_{ik})^2}{2\sigma_{ik}^2}\right) \tag{2}$$

2.3 Random Forest

Random Forest is an ensemble method that builds many decision trees from bootstrap samples and random attribute selection at each split, then combines their predictions through majority voting [15], as in (3), where h is the prediction of each tree out of T trees. This method is relatively resistant to overfitting, requires no distributional assumptions, and can capture nonlinear relationships among attributes; in a large-scale comparative study, the Random Forest family was among the best-performing classifiers [16].

$$\hat{y} = \text{modus}\{h_1(\mathbf{x}), \ldots, h_T(\mathbf{x})\} \tag{3}$$

2.4 Evaluation metrics

Classification performance is measured using a confusion matrix, which yields accuracy, precision, recall, and F1-score [17], [18]. In multiclass classification, precision, recall, and F1-score are computed per class c using (4)–(6), then averaged (macro average) so that every class carries equal weight, including minority classes. Accuracy is the proportion of all correct predictions. Because the class distribution in this study is imbalanced, accuracy must be compared against the majority-class baseline [19].

$$\text{Presisi}_c = \frac{TP_c}{TP_c + FP_c} \tag{4}$$

$$\text{Recall}_c = \frac{TP_c}{TP_c + FN_c} \tag{5}$$

$$F1_c = \frac{2 \times \text{Presisi}_c \times \text{Recall}_c}{\text{Presisi}_c + \text{Recall}_c} \tag{6}$$

Research Methodology

3.1 Dataset

The dataset contains 2,055 rows of supporting-course scores from the Computer Systems (SK) and Information Systems (SI) study programs. Each row contains the student ID (NIM), name, study program, supporting course, supporting-course score, attendance (%), project score, midterm (UTS), final exam (UAS), and study club interest. The supporting courses for SK include Computer Networks, Advanced Computer Networks, Computer Network Administration, Advanced Electronics, Sensors and Actuators, and IoT Platforms and Infrastructure, while those for SI include Data Mining, Data Warehouse, Web Design, and Advanced Web Design. Study club interest consists of four classes: IoT and Network Engineering (SK), and Data Science and Web Programming (SI). The data cover 1,079 unique students, mostly from the 2023 cohort (487 students) and the 2024 cohort (530 students).

3.2 Data preprocessing

Data inspection showed no missing values and no duplicate student ID–course pairs. Because a student may have 1 to 6 supporting courses, the data were aggregated per student by averaging the five numerical variables so that each student becomes a single observation. This step prevents data leakage, i.e., the same student appearing in both the training and test sets. Values of 0 were retained because they may reflect students who did not take the exam or did

not pass, and no normalization was applied because both algorithms are not sensitive to attribute scale. The final result is 1,079 observations with five predictor attributes and one class attribute.

3.3 Modeling and evaluation

The models were built in Python using the scikit-learn library [20]. The Naive Bayes model used was GaussianNB with default parameters, while Random Forest used 100 trees and random_state 42. Evaluation was carried out using two schemes. First, stratified 5-fold cross-validation repeated five times (25 folds) to obtain a stable performance estimate [21]. Second, a stratified 80:20 hold-out test, consisting of 863 training and 216 test samples, to produce a confusion matrix. In addition, performance was compared with the majority-class baseline and analyzed separately for each study program because the SK and SI interest classes do not overlap.

Results and Discussion

4.1 Data characteristics

Table 1 shows the class distribution. The Data Science and Web Programming classes dominate the data (477 and 472 students), while IoT and Network Engineering have only 67 and 63 students. This imbalance affects model performance on the minority classes and sets the majority-class baseline at 44.21%.

Table 1. Distribution of students by study club interest class.

Interest class	Study program	Number of students	Percentage
Data Science	SI	477	44.2%
Web Programming	SI	472	43.7%
IoT	SK	67	6.2%
Network Engineering	SK	63	5.8%
Total		1,079	100.0%

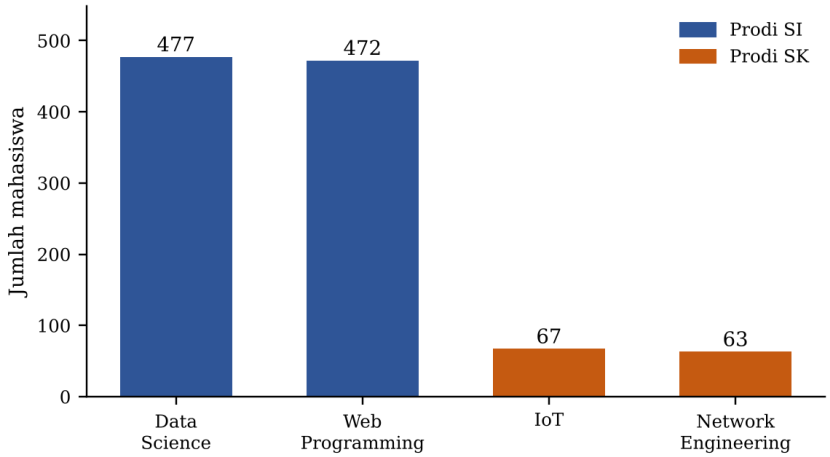


Figure 1. Distribution of study club interest classes.

Table 2 shows the average academic variables per class. Students interested in Data Science and Web Programming have almost identical scores on all variables, so the two classes are hard to distinguish from scores alone. In the SK program, Network Engineering students have lower average midterm and final exam scores than IoT students. The SK classes in general also have lower scores than the SI classes.

Table 2. Average academic variables by interest class.

Interest class	Supporting-course score	Attendance	Project score	Midterm	Final exam
Data Science	80.59	93.55	79.68	78.87	78.72
Web Programming	80.40	93.37	79.05	78.84	77.56
IoT	70.86	90.02	73.47	71.77	72.07

Interest class	Supporting-course score	Attendance	Project score	Midterm	Final exam
Network Engineering	70.88	87.93	69.58	65.75	66.46

4.2 Model performance comparison

Table 3 and Figure 2 present the performance comparison. Based on repeated cross-validation, Random Forest achieved an accuracy of 47.99% and a macro F1-score of 0.395, while Naive Bayes achieved 40.35% and 0.242. The differences are 7.64 percentage points in accuracy and 0.154 in macro F1-score. In the hold-out test, the same pattern emerged (Random Forest accuracy 48.61% and Naive Bayes 39.81%). This result is consistent with earlier findings that Random Forest tends to outperform Naive Bayes on student score data [10], [9] and in specialization prediction [7].

Table 3. Performance comparison of Naive Bayes and Random Forest (mean ± standard deviation across folds in cross-validation).

Model	Accuracy	Macro precision	Macro recall	Macro F1
Naive Bayes (CV)	0.404 ± 0.015	0.288 ± 0.045	0.283 ± 0.030	0.242 ± 0.038
Random Forest (CV)	0.480 ± 0.033	0.468 ± 0.063	0.371 ± 0.028	0.395 ± 0.034
Naive Bayes (hold-out)	0.398	0.289	0.274	0.223
Random Forest (hold-out)	0.486	0.394	0.376	0.382
Majority-class baseline	0.442	–	–	–

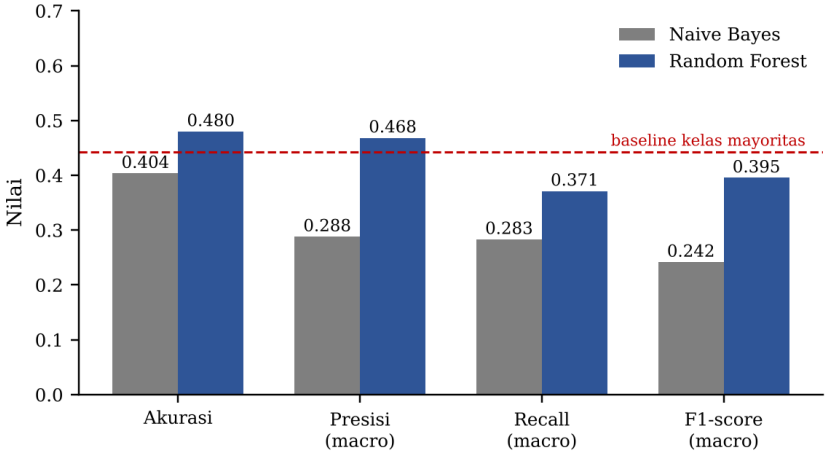


Figure 2. Performance comparison of Naive Bayes and Random Forest (repeated cross-validation).

The accuracy of Naive Bayes (40.35%) is below the majority-class baseline (44.21%), while Random Forest exceeds it by only 3.78 percentage points. This means that academic scores provide limited information for predicting interest. The low performance of Naive Bayes can be explained by violations of its assumptions: attendance, project score, midterm, and final exam are strongly correlated with one another (Pearson correlation coefficients of 0.63–0.87), and their distributions are skewed, with many values in the 75–100 range and a number of 0 values, so the normality and independence assumptions are not met. Random Forest does not depend on these assumptions and can model interactions among attributes.

4.3 Per-class analysis

The confusion matrix in Figure 3 and Table 4 show the behavior of both models per class. Naive Bayes tends to predict almost all SI students as Data Science (Data Science recall 0.844, but Web Programming only 0.021). Random Forest is more balanced across the two SI classes (F1 of about 0.49–0.52), but often confuses Data Science with Web Programming, which is consistent with the nearly identical average scores of the two classes. Random Forest successfully recognizes Network Engineering (F1 0.522) because its midterm and final exam scores are lower, but fails to recognize IoT (F1 0.000). It should be noted that the test data for

the IoT and Network Engineering classes comprise only 13 students per class, so the results must be interpreted with caution.

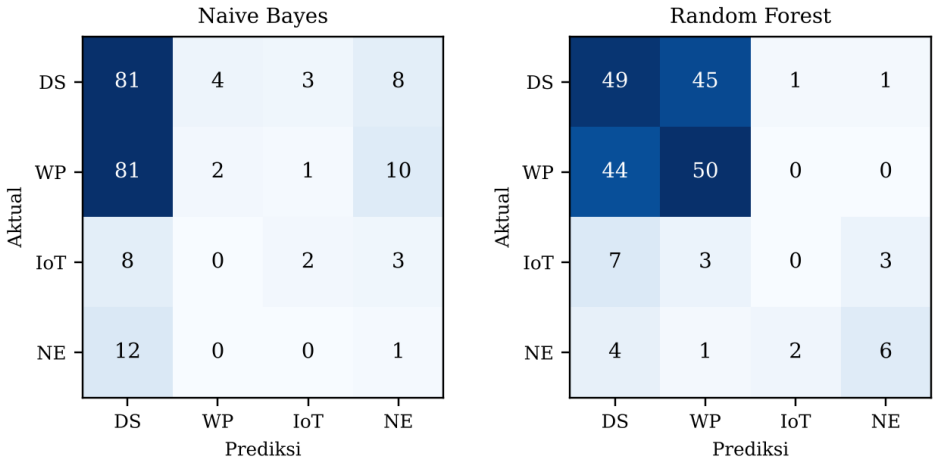


Figure 3. Confusion matrix on the hold-out test (DS: Data Science, WP: Web Programming, NE: Network Engineering).

Table 4. Per-class performance on the hold-out test (precision / recall / F1).

Class	Naive Bayes	Random Forest	Test data
Data Science	0.445 / 0.844 / 0.583	0.471 / 0.510 / 0.490	96
Web Programming	0.333 / 0.021 / 0.040	0.505 / 0.532 / 0.518	94
IoT	0.333 / 0.154 / 0.211	0.000 / 0.000 / 0.000	13
Network Engineering	0.045 / 0.077 / 0.057	0.600 / 0.462 / 0.522	13

4.4 Analysis by study program and important variables

Because the SI and SK classes do not overlap, the models were also evaluated separately (Table 5). For SI (two classes, baseline 50.26%), Random Forest achieved an accuracy of 51.82% and Naive Bayes 47.90%. For SK (two classes, baseline 51.54%), Random Forest achieved 57.69% and Naive Bayes 51.54%, equal to the baseline. Thus, Random Forest remains consistently better, with its advantage over the baseline clearest in SK (6.15 points), while in SI it is only 1.56 points. The standard deviation for SK is relatively large (about 8.8 points) because there are only 130 students.

Table 5. Accuracy and macro F1-score by study program (repeated cross-validation).

Study program	n	Baseline	NB accuracy	RF accuracy	NB F1	RF F1
SI (Data Science vs Web Programming)	949	50.26%	47.90%	51.82%	0.403	0.517
SK (IoT vs Network Engineering)	130	51.54%	51.54%	57.69%	0.464	0.569

Based on the Random Forest feature importance (Figure 4), the supporting-course score is the most important variable (0.317), followed by the final exam (0.184), midterm (0.174), project score (0.164), and attendance (0.162). It should be noted that the contribution of these variables partly reflects the difference in score levels between SK and SI students, not solely differences in interest for each class.

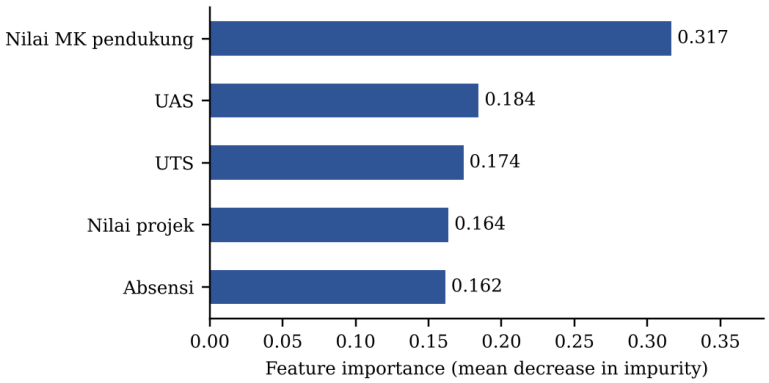


Figure 4. Random Forest feature importance.

4.5 Implications and limitations

The results show that academic scores are useful as a supporting signal, but are not adequate as the sole basis for study club recommendations, especially for distinguishing Data Science from Web Programming. Several limitations should be noted. First, averaging per student discards course-level score information. Second, the IoT and Network Engineering classes are very small, so their performance estimates are unstable. Third, the model parameters were not tuned (hyperparameter tuning) and class imbalance was not addressed with techniques such as SMOTE [22]. Fourth, the data come from a single institution, so generalization needs further testing. Future work can add interest variables (e.g., questionnaire scores), use per-course scores as separate features, and apply class imbalance handling and parameter tuning.

Conclusion

This study compared Naive Bayes and Random Forest for predicting students' study club interest based on academic scores for 1,079 students. Random Forest was superior on all metrics, with an accuracy of 47.99% and a macro F1-score of 0.395, compared with 40.35% and 0.242 for Naive Bayes. This advantage was consistent in the hold-out test and in the per-study-program analysis. However, the accuracy of Random Forest is only slightly above the majority-class baseline (44.21%), so academic scores alone are not strong enough to predict interest. The supporting-course score is the most influential feature. Consequently, a study club recommendation system should combine academic scores with student interest data; further development may include adding interest variables, handling class imbalance, and tuning hyperparameters.

References

[1] Munawirah and A. O. Arisha, “Implementasi Naïve Bayes untuk klasifikasi peminatan program studi pada penerimaan mahasiswa baru di Fakultas Ilmu Komputer Unika,” Bull. Inf. Technol. (BIT), vol. 6, no. 3, pp. 218–229, Sep. 2025.

[2] R. S. J. d. Baker and K. Yacef, “The state of educational data mining in 2009: A review and future visions,” J. Educ. Data Min., vol. 1, no. 1, pp. 3–17, 2009.

[3] C. Romero and S. Ventura, “Educational data mining: A review of the state of the art,” IEEE Trans. Syst., Man, Cybern. C, Appl. Rev., vol. 40, no. 6, pp. 601–618, Nov. 2010.

[4] A. M. Shahiri, W. Husain, and N. A. Rashid, “A review on predicting student’s performance using data mining techniques,” Procedia Comput. Sci., vol. 72, pp. 414–422, 2015.

[5] P. Cortez and A. Silva, “Using data mining to predict secondary school student performance,” in Proc. 5th Future Bus. Technol. Conf. (FUBUTEC), Porto, Portugal, 2008, pp. 5–12.

[6] B. Q. Husaini and J. Jemakmun, "Penerapan algoritma decision tree C45 untuk klasifikasi penjurusan siswa," J. Teknol. Inf. dan Komput. MH Thamrin, vol. 9, no. 1, pp. 455–470, Mar. 2023, doi: 10.37012/jtik.v9i1.1512.
[7] R. W. Tamrin, "Prediksi peminatan mahasiswa dengan algoritma berbasis machine learning (studi kasus: Sistem Informasi Universitas Multimedia Nusantara)," Undergraduate thesis, Universitas Multimedia Nusantara, Tangerang, Indonesia, 2025.
[8] F. Mahardika, F. Supriadi, and A. Guntara, "Analisis distribusi minat mahasiswa pada konsentrasi informatika menggunakan pendekatan data-driven decision making," Inti Nusa Mandiri, vol. 19, no. 2, pp. 278–287, Feb. 2025, doi: 10.33480/inti.v19i2.6347.
[9] B. Yusuf, M. Qalbi, Basrul, I. Dwitawati, Malahayati, and M. Ellyadi, "Implementasi algoritma Naive Bayes dan Random Forest dalam memprediksi prestasi akademik mahasiswa Universitas Islam Negeri Ar-Raniry Banda Aceh," Cyberspace: J. Pendidik. Teknol. Inf., vol. 4, no. 1, pp. 50–58, Mar. 2020.
[10] I. M. B. Adnyana, "Penerapan teknik klasifikasi untuk prediksi kelulusan mahasiswa berdasarkan nilai akademik," J. Teknol. Inf. dan Komput., vol. 7, no. 3, pp. 480–485, Oct. 2021.
[11] F. I. K. Budiyanto, I. Hermadi, and M. K. D. Hardhienata, "Prediksi performa akademik mahasiswa untuk kelulusan predikat cum laude dengan pendekatan machine learning," J. Ilmu Komput. dan Agri-Inform., vol. 11, no. 1, pp. 39–49, 2024, doi: 10.29244/jika.11.1.39-49.
[12] J. Han, M. Kamber, and J. Pei, Data Mining: Concepts and Techniques, 3rd ed. Waltham, MA, USA: Morgan Kaufmann, 2012.
[13] I. Rish, "An empirical study of the naive Bayes classifier," in IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence, 2001, vol. 3, no. 22, pp. 41–46.
[14] P. Domingos and M. Pazzani, "On the optimality of the simple Bayesian classifier under zero-one loss," Mach. Learn., vol. 29, no. 2–3, pp. 103–130, 1997.
[15] L. Breiman, "Random forests," Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001.
[16] M. Fernández-Delgado, E. Cernadas, S. Barro, and D. Amorim, "Do we need hundreds of classifiers to solve real world classification problems?," J. Mach. Learn. Res., vol. 15, pp. 3133–3181, 2014.
[17] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," Inf. Process. Manag., vol. 45, no. 4, pp. 427–437, 2009.
[18] A. Tharwat, "Classification assessment methods," Appl. Comput. Informatics, vol. 17, no. 1, pp. 168–192, Jan. 2021, doi: 10.1016/j.aci.2018.08.003.
[19] H. He and E. A. Garcia, "Learning from imbalanced data," IEEE Trans. Knowl. Data Eng., vol. 21, no. 9, pp. 1263–1284, Sep. 2009.
[20] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," J. Mach. Learn. Res., vol. 12, pp. 2825–2830, 2011.
[21] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in Proc. 14th Int. Joint Conf. Artif. Intell. (IJCAI), Montreal, QC, Canada, 1995, pp. 1137–1143.
[22] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," J. Artif. Intell. Res., vol. 16, pp. 321–357, 2002.