Analyzing Gaming the System Among Novice Programmers: a context of inequality

Hemilis Joyse Barbosa Rocha, PPGI-UFAL, IFAL, hemilis.rocha@ifal.edu.br, https://orcid.org/0000-0001-5422-9271

Evandro de Barros Costa, PPGI-IC-UFAL, evandro@ic.ufal.br, https://orcid.org/0000-0003-4663-8715

Patricia Cabral de Azevedo Restelli Tedesco, Cin-UFPE, pcart@cin.ufpe.br, https://orcid.org/0000-0001-9450-9219

Resumo: *Gaming the System* (GTS) é um comportamento no qual estudantes buscam avançar em ambientes educacionais, explorando regularidades do sistema, em vez de se engajar apropriadamente no processo de aprendizagem. Embora amplamente investigado, o fenômeno permanece pouco compreendido no domínio de programação. Este artigo apresenta um estudo sobre o comportamento de GTS entre estudantes iniciantes em programação matriculados em uma escola pública rural do Nordeste brasileiro. O estudo identificou padrões de comportamento de GTS, desenvolveu e avaliou algoritmos de aprendizado de máquina, como detector automático, e analisou a associação entre GTS e características demográficas, com ênfase em gênero. Os resultados revelaram múltiplos padrões de GTS, incluindo comportamentos ainda não reportados na literatura para o domínio de programação. O algoritmo CART apresentou o melhor desempenho geral, demonstrando viabilidade para aplicação em contextos reais de ensino. A análise demográfica revelou associação estatisticamente significativa entre gênero e a ocorrência de GTS, sugerindo que fatores contextuais e de desigualdade estrutural influenciam a manifestação do fenômeno. Esses achados contribuem para a compreensão do GTS na Educação em Computação, evidenciam a necessidade de abordagens de detecção sensíveis ao contexto sociodemográfico e fornecem subsídios para intervenções pedagógicas voltadas à redução de comportamentos improdutivos em populações historicamente sub-representadas na área.

Palavras-chave: Gaming the System, Programador Novato, Aprendizado de Máquina, Fatores Demográficos.

Analyzing Gaming the System Among Novice Programmers: A context of inequality

Abstract: Gaming the system (GTS) is a behavior in which students seek to advance through educational environments by exploiting system regularities, rather than adequately engaging with the learning process. Although widely investigated, the phenomenon remains little understood in programming education. This paper presents a study about GTS behavior among novice programmers enrolled in a public rural school in Northeastern Brazil. The study identified GTS behavioral patterns exhibited by students, developed and evaluated machine learning algorithms, as an automatic detector for this behavior, and analyzed the association between GTS and demographic characteristics, with particular emphasis on gender. The results revealed multiple GTS behavioral patterns, including patterns not previously reported in the programm ing education literature. CART algorithm achieved the best overall performance, de monstrating viability for deployment in real instructional contexts. Demographic a nalysis revealed a statistically significant association between gender and GTS oc currence, suggesting that contextual and structural inequality factors modulate th e manifestation of the phenomenon. These findings contribute to the understanding o f GTS in Computing Education, highlight the need for detection approaches sensitiv e to sociodemographic context, and provide evidence to support pedagogical interve ntions aimed at reducing unproductive learning behaviors among historically underr epresented populations in the field..

Keywords: Gaming the system, Novice Programmer, Machine Learning, Demographic Fact ors.

1. Introduction

In computer-based learning environments, several undesirable student b ehaviors have been documented, particularly unproductive help-seeking (Alev en et al., 2006) and " gaming the system" (GTS) (Baker et al., 2008). GTS r efers to attempts to achieve success in an educational environment by explo iting system features rather than engaging in meaningful learning and apply ing acquired knowledge to solve problems correctly (Baker e Carvalho, 2008; Baker et al., 2004). This phenomenon has been observed across a variety of e ducational contexts and is consistently associated with poorer learning out comes (Cocea; Hershkovitz e Baker, 2009; Baker et al., 2008; Baker; Mitrovi c e Mathews, 2010; Richey et al., 2021). Consequently, researchers have dev oted considerable effort to modeling and automatically detecting GTS behavi ors using both knowledge engineering approaches (Aleven et al., 2006; Walon oski e Heffernan, 2006) and machine learning techniques (Baker et al., 2008 ; Baker; Mitrovic e Mathews, 2010). Within introductory programming courses , understanding GTS behaviors is particularly important due to the cognitiv e demands faced by novice programmers and the growing concern about the pre valence of these behaviors in programming learning environments (Solyst et a l., 2021). Despite growing interest in the topic, the literature remains li mited regarding how GTS manifests specifically in programming education. Pr evious studies have explored the relationship between students' affective s tates, observable behaviors, and learning outcomes, identifying gaming the s ystem
as one of the behaviors associated with lower performance (Rodrigo et al., 2 009). Other research has focused on unproductive help-seeking among novice p rogrammers, proposing taxonomies that include behaviors such as requesting h ints before attempting a task or repeatedly requesting hints within a short period of time (Marwan; Dombe e Price, 2020). Furthermore, investigations c onducted in flipped CS1 courses have examined factors associated with inapp ropriate learning behaviors, including attitudes related to gaming the syst em, finding results that challenge earlier assumptions regarding the influe

nce of prior knowledge on these behaviors (Solyst et al., 2021; Baker et al., 2005).

Although important advances have been made, gaming the system remains insufficiently understood, particularly in introductory programming contexts. Moreover, evidence from educational research suggests that demographic characteristics may influence both learning outcomes and motivational constructs (Childs, 2017; Zeldin; Britner e Pajares, 2008). For instance, studies in mathematics education have shown that the relationship between hint usage and academic performance varies according to school context, with different patterns emerging in urban and rural environments (Karumbaiah; Ocumpaugh e Baker, 2022). Such findings indicate that contextual and demographic factors may also play a relevant role in shaping gaming behaviors. Motivated by these gaps, this study investigates GTS among novice programmers enrolled in a public rural school in Northeastern Brazil, a region historically characterized by significant social and educational inequalities (Oliveira e Jesus, 2020). This context presents a distinctive combination of factors: limited prior technology access, rural school infrastructure, and regional socioeconomic disparities that may shape how students interact with computer-based learning environments and, consequently, how and why GTS manifests. We ask whether the behavioral patterns, detection features, and demographic correlates of GTS in this setting align with or depart from what has been reported in more studied educational contexts.

Accordingly, the study addresses the following research questions: RQ1: What GTS behavioral patterns are exhibited by novice programmers in a public rural school context, and how effectively can these patterns be detected automatically using machine learning? RQ2: What demographic characteristics, particularly gender, are associated with GTS behavior among novice programmers in this context, and what do these associations suggest about the role of contextual and structural factors in the manifestation of GTS?

To answer these questions, we conducted an observational study involving novice programmers from a public rural school in Northeastern Brazil, collecting interaction data during programming activities. The main contributions of this work are threefold. First, we identify and characterize GTS behavioral patterns exhibited by novice programmers, including patterns not previously reported in programming education contexts. Second, we develop and evaluate an automatic detection model based on three machine learning algorithms: Decision Tree (CART), K-Nearest Neighbors (KNN), and Multilayer Perceptron (MLP), with CART achieving the best overall performance.

2. Gaming the system behavior

Research related to detecting unwanted system abuse behavior has mainly focused on identifying and characterizing these behaviors. However, it is critical to recognize that, although similar to cheating, abusive behavior is n ot necessarily identical to cheating. While abusing the system involves taking advantage of flaws or loopholes present in a system, cheating involves directly violating established rules.

Therefore, it is crucial to understand that students who exhibit these behaviors should not be labeled simply as "abusers" or "cheaters." They have complex motivations that lead them to act in unproductive or unwanted ways (Baker et al., 2004). To better understand these motivations, it is necessary to investigate who these students are and what the origins of their behaviors are. This means exploring the underlying motivations that influence your actions and behaviors within the educational context.

One of the most recent works on variable discovery for detection proposes a latent variable model for detecting " gaming the system" behavior based on item response theory (IRT-GD) in the algebra domain (Huang et al., 2022). The model estimates a latent " gaming" tendency for each student, taking into account contextual factors. These factors include the frequency with which students play in less common gaming contexts compared to more common contexts, as observed by students' action patterns and the predictability of human labels. The IRT-GD is based on a previously validated knowledge engineering game detector (KE-GD), which focuses on students' action characteristics and the predictability of human labels. However, although the study does not explicitly explain the variables considered to implement the detector, the authors claim to have used a previously validated knowledge engineering gaming detector containing 13 interpretable patterns that model (Paquette; Carvalho e Baker, 2014; Paquette e Baker, 2019) behavior.

3. Research Methods

3.1. Materials

The ADA(https://ada-projeto.vercel.app/) is a web-based learning environment developed to support introductory programming courses. The environment was designed to engage students in structured problem-solving activities while collecting detailed interaction logs that enable the analysis of learning behaviors, including gaming the system. It is structured around the following elements: class, sessions, problems, alternatives, attempts, tips, code, and subjects. A class is associated with some students and problem-solving sessions. Each problem has four tips and alternatives related to it. Each student can make several attempts to solve the problems.

The environment has several essential features to help students learn programming. One of these resources is the problem solver, which allows the student to present partial or complete solutions for each proposed problem. When submitting partial solutions, students answer specific questions about solving the problem, while when submitting complete solutions, they have the o pportunity to write programming code to solve the problems. Another important feature of ADA is its tiered tips system.

The tips are organized into different levels, with each level offering progressively specific suggestions on how to proceed at a particular stage of the problem. The structure of the tips starts with more abstract levels and progresses to more specific tips until the answer is direct. Students can request hints at any point during the problem. All these aspects enable the extraction of behavioral features for machine learning-based GTS detection.

3.2. Study Design

This study was conducted in the first semester of 2023 and followed an observational design, in which students' natural interaction behaviors during programming activities were recorded and analyzed without experimental manipulation of any independent variable. The study took place in a public rural technical high school located in the interior of Northeastern Brazil, a region historically characterized by significant social and educational inequalities. Participants were students from two second-year technical high school classes enrolled in a Java programming course (CS2). CS2 students were chosen because they were new to programming and had some familiarity with basic concepts. The study was carried out at this time, as the students were about to start learning object-oriented programming in the CS2 course.

3.2.1. Participants

The study involved 67 students who participated in all problem-solving sessions: 29 from the afternoon class and 38 from the morning class. Of these participants, 35 identified as female and 32 as male, aged between 16 and 21 years. Of these, 61% were white and 39% were of African descent.

3.2.2. Procedure

The study took place during computer programming classes (sessions). Data collection was carried out over four sessions organized into three phases: pre-test, problem-solving intervention, and post-test. In the pre-test session (Session 1), students participated in a 30-minute introductory presentation in which the study objectives were explained, consent forms were distributed, and a comprehensive walkthrough of the ADA environment's features was provided. The remaining 70 minutes of the session were devoted to a problem-solving activity that served as a baseline assessment of students' initial performance.

The intervention phase comprised two additional problem-solving sessions (Sessions 2 and 3), each lasting 100 minutes. Students worked individually, one per computer, in two computer laboratories. During these sessions, three researchers were present in each laboratory and conducted structured behavioral observations. Observers circulated continuously through the lab, codin g each student' s behavior multiple times per session. Specifically, observers recorded the occurrence and nature of GTS behavior following the observation protocol established in prior work (Baker e Carvalho, 2008; Paquette; Carvalho e Baker, 2014; Baker; Corbett e Koedinger, 2004). Each observer independently coded the frequency and behavioral category of GTS manifestations, and inter-rater agreement was assessed.

3.3. Data

Through the study detailed in the previous section, we collected a dataset comprising 4,492 records of student interactions within the learning environment. A total of 4,492 instances were included in the dataset. Of these, 2,814 instances (62.63%) were classified as showing no evidence of "gaming the system" behavior (Class 0), while 1,678 instances (37.37%) were classified as showing evidence of "gaming the system" behavior (Class 1). . The labeling process utilized a model based on thirteen previously validated patterns of " gaming the system" behavior (Paquette; Carvalho e Baker, 2014), ensuring the reliability and consistency of the dataset annotations.

However, to use this model it was necessary to understand the concept of "text replays" , which are groupings of up to five sequences of student actions when interacting with the environment. Thus, if a pattern is identified in the analysis of the data set, all actions involved are labeled as "gaming the system" . These text repetitions facilitate efficient classification and have acceptable reliability (Paquette; Carvalho e Baker, 2014), and have also been previously used to train detectors of the same behavior. In the first labeling step, coders had a relatively low inter-rater agreement, reflected by a Cohen' s Kappa coefficient of 0.42. To improve consistency, a second round of labeling was carried out, where divergences were analyzed and discussed, leading to a more aligned interpretation of behaviors. This collaborative approach resulted in a significant increase in Cohen' s Kappa coefficient to 0.71. However, it was still considered necessary to improve the agreement. Therefore, a third round was carried out to resolve the remaining conflicts, culminating in a very satisfactory final coefficient of 0.93, indicating a high level of agreement between coders.

After annotating our data set, we conducted a three-stage analysis: pre-processing, classification, and data evaluation. In pre-processing, we used a technique called cross-validation. This is a way of training and testing the model several times on different data, which increases the algorithm' s ability to generalize and makes the predictions more accurate (Browne, 2000). For classification, we employ three algorithms: decision tree, k-nearest neighbors (KNN), and Multilayer Perceptron (MLP) neural networks, based on previous studies such as those by Baker et al. (2008). The implementation was carried out using the scikit-learn library. The decision tree was configured with the CART algorithm, KNN with 3 neighbors, and MLP neural network. For e valuation, we analyzed the performance of each algorithm on the training and testing subsets using six metrics: area under the ROC curve (AUC), accuracy, precision, recall, F1 score, and Kappa. A significance level of $p < 0.05$ was applied. AUC evaluates the quality of the model's prediction, while precision, recall, and F1 provide a comprehensive performance assessment. Kappa ensures that the model's correct predictions are not random. Accuracy is an evaluation metric that measures the overall correctness of a classification model.

4. Results and Discussion

This section presents and discusses the results organized around the two research questions defined in Section 1. As mentioned in Section 3.3, we used the patterns described in the model of a previous work to annotate the data (Paquette; Carvalho e Baker, 2014), reporting the GTS behavioral patterns identified through observation. However, when we analyzed the application of this model to the data and functionalities of the ADA environment, we found that it was not compatible with some standards. This incompatibility stems from the difference between the types of tasks offered by the environment in which the model was developed and the problem-solving tasks available in the ADA environment. Despite this, as the patterns described in the previous work are actually data annotation rules, we realized that patterns 1, 5, 6 and 11 were suitable for our environment. We therefore used them to annotate the data, and analyze performance in our data context and performance of our classifiers. In what follows, Section 4.1 addresses RQ1, discussing the performance of the three machine learning algorithms in automatic GTS detection. Section 4.2 addresses RQ2, presenting the demographic characteristics associated with GTS behavior, with particular emphasis on gender. Where applicable, findings are interpreted in relation to prior work and discussed in terms of their implications for programming education in socioeconomically vulnerable contexts. .

4.1. RQ 01: Automatic GTS Detector

To address our first research question, regarding the effectiveness of the proposed automatic detector in identifying GTS behavior among novice programmers, we trained and evaluated three machine learning algorithms (Decision Tree using CART, KNN, and MLP) on the annotated interaction data from the ADA environment. The algorithms were evaluated across the four GTS patterns selected for this context. The following two tables present the performance of each algorithm across 6 metrics: Accuracy, precision, recall, F-measure (or F1-Score), Cohen's Kappa, and ROC.

Table 1. Results of comparing algorithms in training

Algoritm	ACU	PRE	REC	FMe	KAP	ROC

DOI:

KNN	0,84 ±0,002	0,84 ±0,002	0,84 ±0,002	0,85 ±0,002	0,69 ±0,004	0,84 ±0,002
CART	1,00 ± 0,00	1,00 ± 0,00	1,00 ± 0,00	1,00 ± 0,00	1,00 ± 0,00	1,00 ± 0,00
MLP	0,75 ±0,006	0,75 ±0,005	0,75 ±0,005	0,76 ±0,004	0,51± 0,011	0,75± 0,006

Table 2. Results of comparing algorithms in test

Algoritm	ACU	PRE	REC	FMe	KAP	ROC
KNN	0,69 ±0,008	0,69 ±0,008	0,69 ±0,008	0,70 ±0,008	0,39 ±0,016	0,69 ±0,008
CART	0,88 ±0,007	0,88 ±0,007	0,88 ±0,007	0,88 ±0,007	0,76 ±0,014	0,88 ±0,007
MLP	0,71 ±0,005	0,71 ±0,005	0,71 ±0,005	0,72 ±0,010	0,40 ±0,010	0,71 ±0,005

Tables 1 and 2 provide an analysis of the performance of three classification algorithms: KNN, CART and MLP, on two distinct data sets: training and testing. In the training set, the results indicate that the CART algorithm achieved a perfect accuracy of 1.00, while the KNN achieved an average accuracy of 0.84 and the MLP of 0.75. Furthermore, all metrics, including precision, recall, F-Measure, Kappa, and area under the ROC curve, reflected high performance for the CART algorithm, with values consistently close to 1.00. However, both KNN and MLP also showed satisfactory results in these metrics.

In the test set, the results show a similar trend, with the CART algorithm maintaining a high average accuracy of 0.88, followed by the MLP with 0.71 and the KNN with 0.69. Likewise, the metrics of precision, recall, F-Measure, Kappa and area under the ROC curve indicated consistent performance of the algorithms, with CART again leading in all categories. In summary, the results suggest that the CART algorithm performed best on both datasets, followed by MLP and finally KNN. Using the Wilcoxon test, we compared the performance of the algorithms at a significance level of $\alpha = 0.05$. In the training phase, we observed a significant difference between the performance of CART and KNN ($p = 0.022$), while there was no significant difference between CART and MLP ($p = 0.061$). In the test phase, there are significant differences between CART versus MLP ($p = 0.021$) and CART versus KNN ($p = 0.021$). The

DOI:

refore, the CART decision tree algorithm was the best performer, which is why we will analyze the performance of the decision tree on all patterns.

Table 3. Results obtained when applying the decision tree algorithm to the "gaming the system" pattern (Training)

Pattern	ACU	PRE	REC	FMe	KAP	ROC
PA01	0.80 ± 0.02	0.75 ± 0.05	0.61 ± 0.07	0.67 ± 0.09	0.53 ± 0.04	0.75 ± 0.01
PA05	0.80 ± 0.01	0.73 ± 0.03	0.62 ± 0.04	0.67 ± 0.01	0.54 ± 0.02	0.75 ± 0.01
PA06	0.79 ± 0.01	0.72 ± 0.05	0.64 ± 0.03	0.68 ± 0.01	0.53 ± 0.04	0.76 ± 0.01
PA11	0.85 ± 0.04	0.81 ± 0.03	0.67 ± 0.08	0.69 ± 0.03	0.57 ± 0.02	0.83 ± 0.01

Table 4. Results obtained when applying the decision tree algorithm to the "gaming the system" pattern (Test)

Pattern	ACU	PRE	REC	FMe	KAP	ROC
PA01	0.64 ± 0.01	0.45 ± 0.03	0.34 ± 0.03	0.39 ± 0.02	0.14 ± 0.07	0.57 ± 0.02
PA05	0.61 ± 0.01	0.41 ± 0.01	0.40 ± 0.06	0.41 ± 0.01	0.12 ± 0.03	0.56 ± 0.02
PA06	0.59 ± 0.02	0.39 ± 0.04	0.38 ± 0.06	0.39 ± 0.02	0.08 ± 0.01	0.54 ± 0.02
PA11	0.72 ± 0.01	0.37 ± 0.05	0.42 ± 0.03	0.59 ± 0.04	0.35 ± 0.01	0.67 ± 0.02

Given that the decision tree achieved the best performance, we evaluated the performance of this algorithm on each pattern. Tables 3 and 4 present the results of applying the decision tree algorithm to different behavior patterns, both in the training data set and in the test set. In the training set, we observed that all patterns (PA01, PA05, PA06 and PA11) had a very reasonable average accuracy, ranging from 0.79 to 0.85. Standard PA11 had the highest average accuracy, closely followed by PA01 and PA05, while PA06 had the lowest average accuracy among the standards analyzed. Regarding precision, all patterns consistently presented values above 0.70, indicating a good rate of true positives in relation to the total number of positive predicti ons. However, we observed a variation in recall metrics, F-Measure, Kappa and area under the ROC curve between the different patterns, suggesting different degrees of the model’ s ability to correctly identify positive and negative instances, as well as its overall classification ability. In the test set, the results vary more significantly between the different standards. While PA11 maintains a good performance, with an average accuracy of 0.72, the other standards (PA01, PA05 and PA06) show a drop in their performance metrics in relation to the training set. This suggests that the model may be experiencing difficulties in generalizing its learning to new data, especially regarding less representative patterns. The Kappa metric and the area under the ROC curve also reflect this trend, indicating a reduction in agreement between model predictions and true labels.

Comparing the results of this research with previous work, in previous work the authors state that the complete model was applied to the training set, where it accurate detected 340 (64.03%) game clips, misdiagnosed 551 (7.07%) non-game clips, and obtained a Kappa coefficient of 0.430 (Paquette; Carvalho e Baker, 2014). In the test set, the model achieved an accuracy of 93 (52.54%) gaming clips, while misdiagnosing 210 (8.67%) non-gaming clips, resulting in a Kappa coefficient of 0.330. This performance was comparable to Baker’ s (Baker e Carvalho, 2008) model. In this research, the best algorithm evaluated was the decision tree CART, achieving a hit rate between 76% and 100% in all the metrics evaluated.

4.2. RQ 02: Demographic context

In this section, we present the results of the analysis of three demographic variables: gender, age, and race/ethnicity. To analyze the age variable, three age groups were created: group 1 (16-17 years old), group 2 (18-19 years old), and group 3 (20-21 years old). Figure 1 shows the percentage of students who exhibited gaming behavior according to age group. The highest proportion of gaming behavior was observed among students aged 16-17 years. The results revealed that students in group 1 engaged in gaming behavior significantly more often than students in group 2 (($p<0.05$)) and group 3 (($p<0.05$)). Furthermore, students in group 2 also exhibited higher levels of gaming behavior than those in group 3 ((p!0.05)). These findings suggest that younger novice programmers are more likely to engage in gaming the system behaviors. This result is consistent with previous research indicating that motivational and self-regulatory skills tend to develop with age and educational experience, reducing the likelihood of maladaptive learning strategies (Childs, 2017; Zeldin; Britner e Pajares, 2008). Additionally, younger students may have less experience dealing with academic frustration and may therefore be more inclined to exploit system features rather than persist in problem-solving activities.

Table 1 presents the frequency of gaming behavior according to race/ethnicity. The highest percentage of gaming behavior was observed among students who self-identified as White or Asian (61%). The Mann-Whitney test indicated that this group engaged in gaming behavior significantly more frequently than the other ethnic groups ($p < 0.05$). Although studies specifically examining race/ethnicity and gaming the system are scarce, previous research has shown that demographic and contextual characteristics can influence how students interact with educational technologies (Karumbaiah; Ocumpaugh e Baker, 2022). Differences in help-seeking patterns, motivation, and engagement have been reported across demographic groups, suggesting that contextual and sociocultural factors may shape students' learning strategies. Therefore, our findings reinforce the importance of considering demographic characteristics when investigating gaming behavior, particularly in underrepresented educational settings such as rural schools in developing regions.

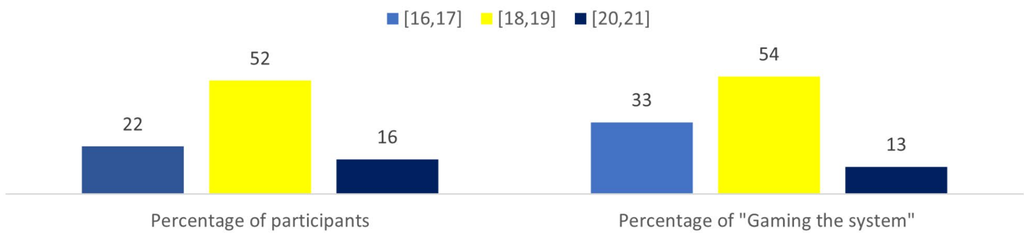

Figure 1 - Percentage of "gaming" by demographic variable age group.

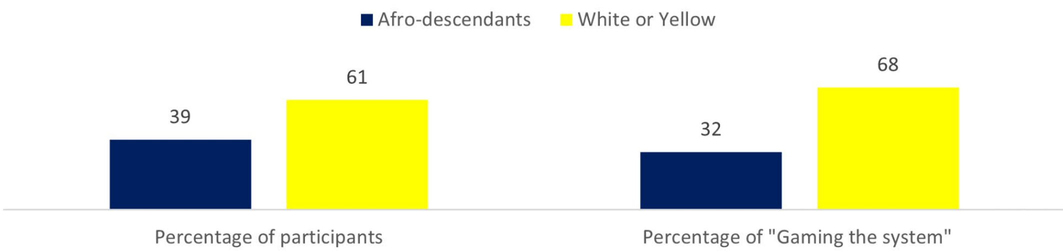

Figure 2 - Percentage of "gaming" by ethnic-racial demographic variable.

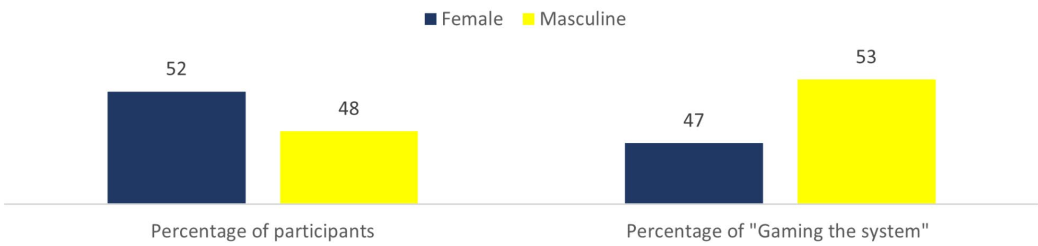

Figure 3 - Percentage of "gaming" by demographic variable gender.

Table 5 shows the outcomes of the Mann-Whitney test comparing learning gains across groups defined by ethnicity (Afro-descendant and White/Yellow) and gaming behavior (gaming and non-gaming students). Significant differences in learning gains were observed across the analyzed groups. In particular, students who did not engage in gaming the system achieved substantially high er learning gains than those who exhibited such behavior, regardless of their ethnic background. These findings are consistent with Figura 1. Percentage of " gaming" by demographic variable age group Figura 2. Percentage of " gaming" by ethnic-racial demographic variable previous studies reporting that gaming the system is generally associated with lower learning outcomes and reduced knowledge acquisition (Cocea; Hershkovitz e Baker, 2009; Baker et al., 2008; Baker et al., 2008). The results also suggest that the negative impact of gaming behavior may transcend demographic categories, reinforcing the idea that disengagement from productive learning processes is a critical factor affecting student achievement.

Table 5 presents the Mann-Whitney test results examining learning gains according to gender (female and male) and gaming behavior. Significant differences were found between students who engaged in gaming and those who did not, with non-gaming students consistently achieving higher learning gains. Female students who did not exhibit gaming behavior showed particularly strong gains compared to their counterparts who engaged in gaming. These findings align with prior research indicating that motivational and self-regulatory factors often vary across demographic groups and can influence learning outcomes (Zeldin; Britner e Pajares, 2008; Childs, 2017). More importantly, our results reinforce the evidence that gaming the system represents a maladaptive learning strategy associated with poorer academic performance regardless of gender (Cocea; Hershkovitz e Baker, 2009; Baker et al., 2008). Table 6 reports the results for age groups. Students in age group 1 who did not engage in gaming behavior achieved significantly higher learning gains than students from all gaming groups. Similarly, students in age group 3 who avoided gaming the system outperformed both their gaming counterparts and younger students who exhibited gaming behavior. No statistically significant differences were observed for age group 2. These findings suggest that the detrimental effects of gaming the system may be particularly pronounced among younger learners. This interpretation is supported by studies indicating that younger students tend to possess less developed self-regulation and metacognitive skills, making them more vulnerable to ineffective learning strategies (Childs, 2017). Furthermore, the results corroborate previous research showing that gaming the system is negatively associated with learning gains and may hinder the development of deeper conceptual understanding (Baker et al., 2008; Cocea; Hershkovitz e Baker, 2009; Baker; Mitrovic e Mathews, 2010). Taken together, these findings highlight that although demographic characteristics may influence how gaming behavior manifests, the behavior itself remains consistently associated with lower learning gains across different groups.

Table 5. Results of Mann-Whitney test for comparison of learning gain by gender and learning gain by age group and" gaming the system" behavior.

No game / Game	Female	Male	Age group 1	Age group 2	Age group 3	Afrodescendant	White/Yellow
Female	p<0.05	p<0.05	–	–	–	–	–
Male	p< 0.05	p<0.05	–	–	–	–	–
Age group 1	–	–	p=0.05	p < 0.05	p < 0.05	–	–
Age group 2	–	–	p = 0.31	p = 0.35	–	–	–
Age group 3	–	–	p < 0.05	–	–	–	–
Afrodescendant	–	–	–	–	–	p < 0.05	p < 0.05
White/Yellow	–	–	–	–	–	p < 0.05	p < 0.05

Table 2 presents the frequency of gaming the system behavior according to gender. Descriptive statistics indicate differences in the distribution of gaming behavior between female and male students, with 52% of the identified gaming cases occurring among female students. However, the Mann-Whitney test revealed that male students engaged in gaming behavior significantly more frequently than female students ($p < 0.05$). This finding is consistent with previous research suggesting that gender differences may influence motivational, affective, and self-regulatory processes in educational settings (Zeldin; Britner e Pajares, 2008; Childs, 2017). Studies have reported that female students often exhibit higher levels of persistence and constructive help-seeking behaviors, while male students may be more likely to adopt less productive strategies when facing learning difficulties (Aleven et al., 2006). In the context of computing education, affective and behavioral factors have been shown to influence students' engagement and performance during programming activities (Rodrigo et al., 2009). Moreover, previous studies on gaming the system have highlighted the role of motivation and help-seeking patterns in the emergence of this behavior (Baker et al., 2008; Aleven et al., 2006). Therefore, the greater frequency of gaming behavior observed among male students may reflect differences in how learners respond to challenges and interact with support mechanisms available in the learning environment.

5. Conclusion

This study investigated gaming the system behavior among novice programmers enrolled in a public rural school in Northeast Brazil through an observatio nal research design. Machine learning-based detection of GTS proved viable in introductory programming contexts, with CART algorithm achieving the best overall performance, an inherently interpretable model suitable for deployment in real instructional settings. The behavioral analysis revealed multiple GTS patterns, including patterns not previously reported in programming education, extending the characterization of this behavior beyond the domains in which it has been most studied. Demographic analysis revealed statistically significant associations between GTS and gender, age, and ethnic background, reinforcing the argument that contextual and structural factors modulate how GTS manifests and that detection approaches must be sensitive to sociodemographic context. Students who exhibited GTS also achieved lower learning gains than their non-gaming peers (Cocea; Hershkovitz e Baker, 2009; Baker et al., 2008; Baker et al., 2008), strengthening the case for timely and targeted intervention.

The three main contributions of this work are: (i) the identification and characterization of GTS behavioral patterns among novice programmers, including previously unreported patterns in this domain; (ii) the development and evaluation of an automatic detection model based on CART, KNN, and MLP, with CART achieving the best overall performance; and (iii) an in-depth demographic analysis revealing how structural and contextual factors shape GTS manifestation in underrepresented educational settings. As future work, comparative studies involving students from different schools, regions, and socioeconomic contexts are needed to assess the generalizability of the behavioral patterns and detector performance. The novel GTS patterns identified here should be investigated in other programming education contexts, and detection approaches explicitly sensitive to sociodemographic student profiles should be developed.

References

ALEVEN, V.; McLAREN, B. M.; ROLL, I.; KOEDINGER, K. R. Toward meta-cognitive tutoring: a model of help seeking with a cognitive tutor. *International Journal of Artificial Intelligence in Education*, v. 16, p. 101-130, 2006.

BAKER, R. et al. Why students engage in gaming the system behavior in interactive learning environments. *Journal of Interactive Learning Research*, v. 19, n. 2, p. 185-224, 2008.

BAKER, R. S.; CORBETT, A. T.; KOEDINGER, K. R. Detecting student misuse of intelligent tutoring systems. In: INTERNATIONAL CONFERENCE ON INTELLIGENT TUTORING SYSTEMS. [S. l.]: Springer, 2004. p. 531-540.

BAKER, R. S.; CORBETT, A. T.; KOEDINGER, K. R.; WAGNER, A. Z. Off-task behavior in the cognitive tutor classroom: when students game the system. In: PROCEEDINGS OF THE SIGCHI CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS. [S. l.: s. n.], 2004. p. 383-390.

BAKER, R. S.; CORBETT, A. T.; KOEDINGER, K. R.; WAGNER, A. Z. Do performance goals lead students to game the system? In: PROCEEDINGS OF ARTIFICIAL INTELLIGENCE IN EDUCATION. [S. l.: s. n.], 2005.

BAKER, R. S. J. d.; CARVALHO, A. M. J. A. de. Labeling student behavior faster and more precisely with text replays. In: PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON EDUCATIONAL DATA MINING. [S. l.: s. n.], 2008. p. 38-47.

BAKER, R. S. J. d.; CORBETT, A. T.; ROLL, I.; KOEDINGER, K. R. Developing a generalizable detector of when students game the system. *User Modeling and User-Adapted Interaction*, v. 18, n. 3, p. 287-314, 2008.

BAKER, R. S. J. d.; MITROVIC, A.; MATHEWS, M. Detecting gaming the system in constraint-based tutors. In: PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON USER MODELING, ADAPTATION AND PERSONALIZATION. [S. l.: s. n.], 2010. p. 267-278.

BROWNE, M. W. Cross-validation methods. *Journal of Mathematical Psychology*, v. 44, n. 1, p. 108-132, 2000.

CHILDS, D. S. *Effects of Math Identity and Learning Opportunities on Racial Differences in Math Engagement, Advanced Course-Taking, and STEM Aspiration*. 2017. Tese (Doutorado) – Temple University, 2017.

COCEA, M.; HERSHKOVITZ, A.; BAKER, R. S. J. d. The impact of off-task and gaming behaviors on learning: immediate or aggregate? In: PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE IN EDUCATION. [S. l.: s. n.], 2009. p. 507-514.

HUANG, Y. et al. Item response theory-based gaming detection. In: PROCEEDINGS OF THE INTERNATIONAL EDUCATIONAL DATA MINING SOCIETY. [S. l.: s. n.], 2022.

KARUMBAIAH, S.; OCUMPAUGH, J.; BAKER, R. S. Context matters: differing implications of motivation and help-seeking in educational technology. *International Journal of Artificial Intelligence in Education*, v. 32, n. 3, p. 685-724, 2022.

MARWAN, S.; DOMBE, A.; PRICE, T. W. Unproductive help-seeking in programming: what it is and how to address it. In: PROCEEDINGS OF THE ACM CONFERENCE ON INNOVATION AND TECHNOLOGY IN COMPUTER SCIENCE EDUCATION. [S. l.: s. n.], 2020. p. 54-60.

OLIVEIRA, D. A.; JESUS, A. M. C. de. As políticas de avaliação e responsabilização no Brasil: uma análise da educação básica nos estados da região Nordeste. *Revista Ibero-Americana de Educação*, 2020.

PAQUETTE, L.; BAKER, R. S. Comparing machine learning to knowledge engineering for student behavior modeling: a case study in gaming the system. *Interactive Learning Environments*, v. 27, n. 5-6, p. 585-597, 2019.

PAQUETTE, L.; CARVALHO, A. M. J. B. de; BAKER, R. S. Towards understanding expert coding of student disengagement in online learning. In: PROCEEDINGS OF THE ANNUAL MEETING OF THE COGNITIVE SCIENCE SOCIETY. [S. l.: s. n.], 2014.

RICHEY, J. E. et al. Gaming and confusion explain learning advantages for a math digital learning game. In: ARTIFICIAL INTELLIGENCE IN EDUCATION. [S. l.]: Springer, 2021. p. 412-424.

RODRIGO, M. M. T. et al. Affective and behavioral predictors of novice programmer achievement. In: PROCEEDINGS OF ITiCSE. [S. l.: s. n.], 2009. p. 156-160.

SOLYST, J. et al. Procrastination and gaming in an online homework system of an inverted CS1. In: PROCEEDINGS OF THE ACM TECHNICAL SYMPOSIUM ON COMPUTER SCIENCE EDUCATION. [S. l.: s. n.], 2021. p. 1095-1101.

WALONOSKI, J. A.; HEFFERNAN, N. T. Detection and analysis of off-task gaming behavior in intelligent tutoring systems. In: INTERNATIONAL CONFERENCE ON INTELLIGENT TUTORING SYSTEMS. [S. l.]: Springer, 2006. p. 382-391.

ZELDIN, A. L.; BRITNER, S. L.; PAJARES, F. A comparative study of the self-efficacy beliefs of successful men and women in mathematics, science, and technology careers. *Journal of Research in Science Teaching*, v. 45, n. 9, p. 1036-1058, 2008.