Beyond Accuracy: A Statistical Validation of Feature Engineering Methods for League of Legends Match Prediction

J. B. de Aguiar Júnior
Programa de Pós-Graduação em Engenharia da Computação
Universidade de Pernambuco (UPE)
Email: jbaj@ecomp.poli.br

***Resumo*—Predicting outcomes in Multiplayer Online Battle Arena (MOBA) games is a complex pattern recognition task. Current approaches often fail to apply rigorous statistical validation when comparing classifiers. This paper proposes "Cypher's Edge,"a hybrid system (React, Node.js, and Python) for the collection, processing, and predictive analysis of individualized performance. A comparative experiment was conducted between two datasets: the proposed "Cypher's Edge"dataset ($N = 48$) and a "Literature 1"dataset ($N = 500$). Using classifiers (Random Forest, MLP, Naive Bayes) evaluated with 10-Fold Cross-Validation, the Nemenyi hypothesis test was applied. The results demonstrate that, although Random Forest models achieved higher accuracy rankings, **there was no statistically significant difference** between the three classifiers. Furthermore, the feature importance analysis identified **turretTakedowns**, **goldEarned**, and **totalDamageDealtToChampions** as the most determinant attributes for victory.**

***Index Terms*—League of Legends, Machine Learning, Classification, Feature Engineering, Friedman Test, Nemenyi, ROC Curve, Riot Games API.**

I. Introduction

Games of the MOBA (Multiplayer Online Battle Arena) genre, such as League of Legends (LoL), represent a cultural and economic phenomenon that attracts millions of players. The vast complexity of these games makes predicting outcomes a challenging task of great academic interest. Machine learning models have been widely applied to predict the winning team based on statistical data from the matches.

However, an analysis of the literature reveals two main gaps. First, many approaches result in "generalist"models, focused on maximizing global accuracy, but which fail to provide actionable and personalized insights for an individual player. Second, the comparison between algorithms is often limited to accuracy, lacking in-depth statistical rigor, such as hypothesis testing, to validate whether the superiority of one model is significant.

To address these gaps, this paper proposes Cypher's Edge: a player-centric hybrid predictive analysis system (React, Node.js, and Python). The system employs a machine learning pipeline that uses robust hypothesis tests (Friedman and Nemenyi) for classifier comparison and provides an interactive attribute relevance analysis (feature importance) engine.

II. Proposed Method

This work introduces Cypher's Edge, a hybrid system for the collection, processing, and predictive analysis of performance data. The system's architecture (Fig. 1) combines an interactive web platform with a decoupled machine learning pipeline.

A. Hybrid System Architecture

The platform was developed using a full-stack architecture. The **frontend** (React.js) provides the UI. The **backend** (Node.js with Express.js) acts as an orchestrator, managing requests, database access, and triggering analysis scripts. Raw match data is collected via the Riot Games API and stored in a MongoDB database. The **ML component** (Python/Scikit-learn) operates independently and is invoked by the Node.js server.

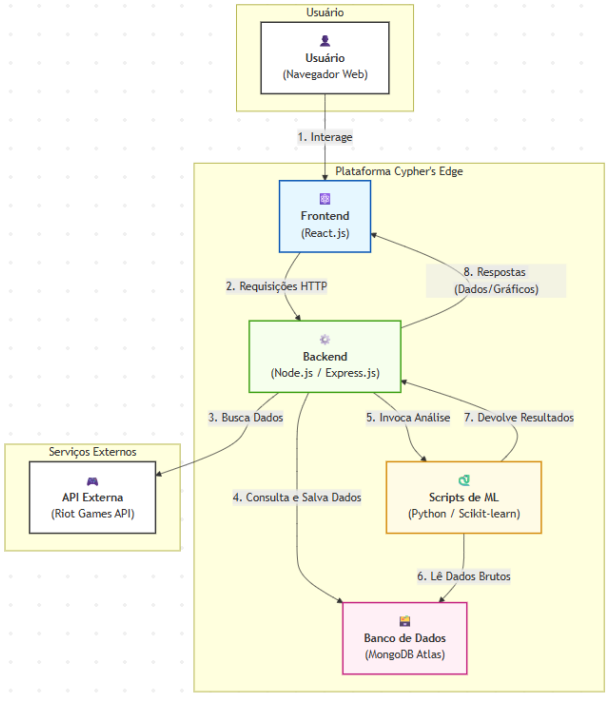


Figura 1: Architecture of the "Cypher's Edge"hybrid system (Fig. 1 from .docx).

B. Predictive Modeling and Evaluation

For the classification task, a multi-classifier (ensemble) approach was employed, as discussed in [1], to compare different learning biases. The selected models were Random Forest (RF), Multilayer Perceptron (MLP), and Naive Bayes (NB), all implemented using the Scikit-learn library [2]. The process (encapsulated in `lol_classifier_model.py`) uses Stratified k-fold Cross-Validation (k=10) to ensure statistical robustness.

III. Experiments and Results

To evaluate the effectiveness of the system, a series of experiments was conducted.

A. Experimental Setup

Database: The experiments used two real datasets extracted via the Riot Games API. The "Cypher's Edge"dataset ($N = 48$) and the "Literatura 1"dataset ($N = 500$), which is an adaptation from the work of [3] and [4]. The task was binary classification (Victory/Defeat).

Compared Systems: For each dataset, three distinct classifiers were trained (Random Forest, MLP, and Naive Bayes). Performance was evaluated using Stratified k-fold Cross-Validation ($k = 10$).

B. Systems Performance Analysis

The performance of the six combinations (2 datasets x 3 classifiers) was compiled. The ranking plot (Fig. 2) shows the average accuracy, where "Cypher's Edge - Random Forest"achieved the highest mean accuracy (Rank 1).

The heatmap (Fig. 3) and the bar chart (Fig. 4) provide a complete comparison of all metrics, confirming the strong performance of Random Forest models across both datasets.

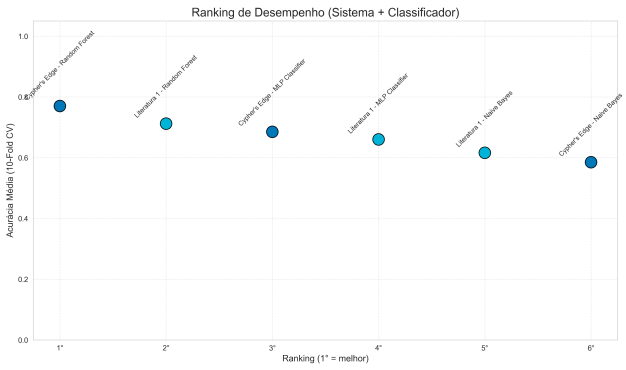


Figura 2: Average performance ranking of evaluated models. Lower rank indicates better performance.

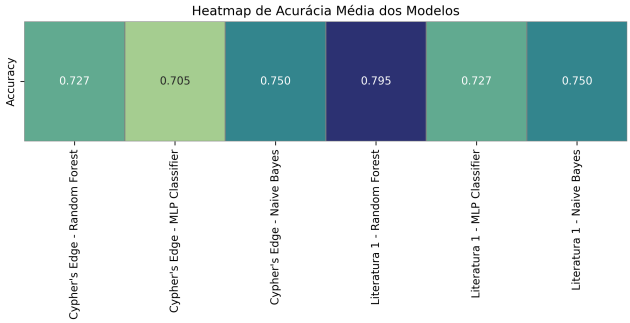


Figura 3: Heatmap of Mean Accuracy for all 6 models.

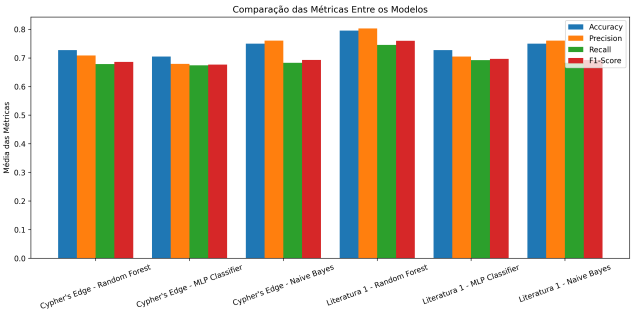


Figura 4: Comparison of Accuracy, Precision, Recall, and F1-Score.

C. Attribute Relevance Analysis

To understand which factors most influenced the predictions of the best-performing model (Random Forest), a feature importance analysis was conducted. Fig. 5 illustrates the most relevant attributes.

The analysis reveals that **`turretTakedowns`** and **`goldEarned`** are the strongest predictors, followed by **`totalDamageDealtToChampions`** and **`totalMinionsKilled`**. This highlights that strategic objectives and economy are more determinant than simple combat metrics like `kills` (ranked 5th).

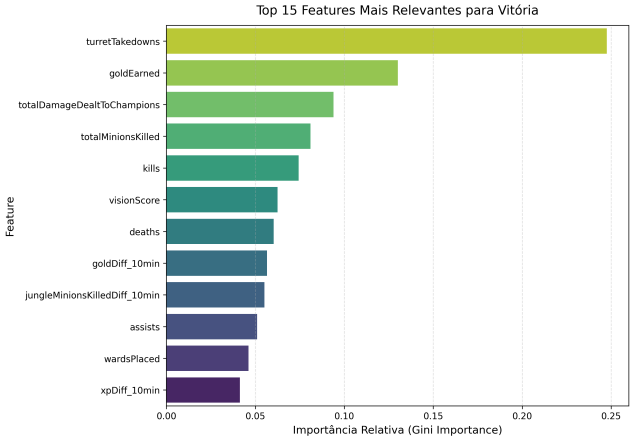


Figura 5: Top-15 most relevant features (Gini Importance).

D. Diagnostic Analysis (ROC and Confusion Matrices)

The performance of the models in the ROC space (Fig. 6) visualizes the trade-off between True Positive Rate (TPR) and False Positive Rate (FPR). The Random Forest and MLP models (circles and 'x') are positioned in the upper-left quadrant (ideal), while the Naive Bayes models (squares) show weaker performance.

This is confirmed by the individual confusion matrices (Fig. 7), which detail the errors (False Positives/Negatives) for each of the six models.

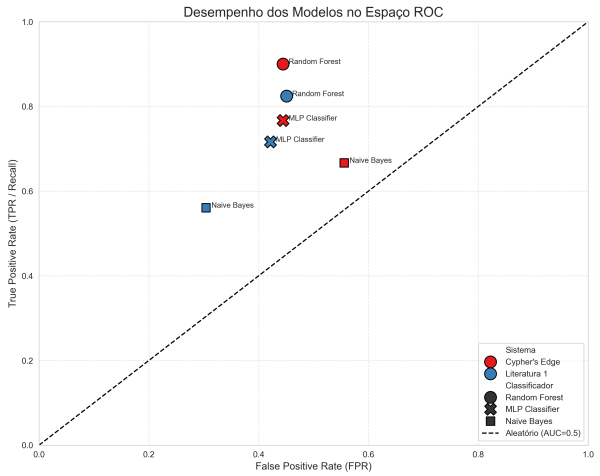


Figura 6: Performance of the 6 models in the ROC Space.

E. Statistical Analysis (Hypothesis Test)

To ensure robust validation, the non-parametric Friedman test was applied, followed by the Nemenyi post-hoc test. This test compares the *classifiers* ($k = 3$) across the *folds* and *datasets* ($N = 20$, from 10 folds x 2 datasets).

The Nemenyi plot (Fig. 9) visualizes the average rankings. The black bar (CD = 0.741) connects methods with no significant statistical difference.

Statistical Finding: The analysis reveals that the Critical Difference (CD) bar connects all three classifiers (Random Forest, MLP Classifier, and Naive Bayes). This indicates that, although Random Forest achieved a better numerical ranking, **there is no statistically significant difference** between the performances of the three models in this experiment ($p = 0.05$).

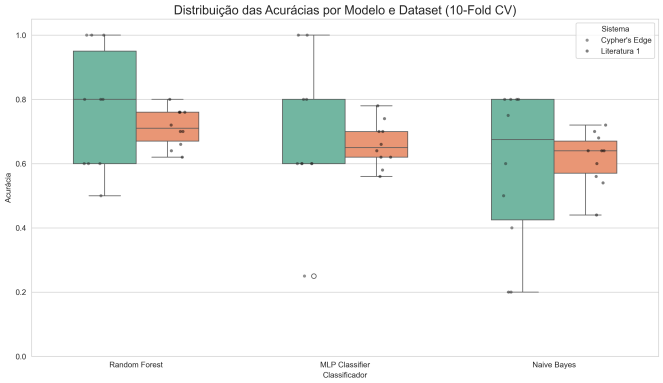


Figura 8: Distribution of 10-Fold CV accuracy scores.

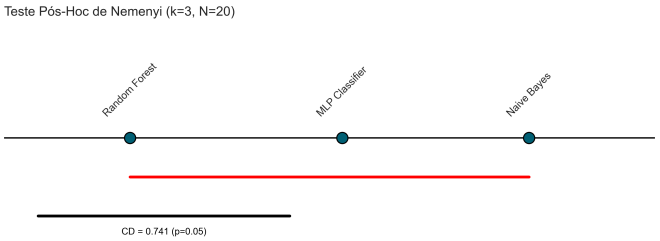


Figura 9: Nemenyi test comparing the rank of the 3 classifiers (k=3, N=20, alpha=0.05).

IV. Conclusion

The objective of this work was to propose and implement "Cypher's Edge,"a hybrid system for predictive analysis, addressing gaps in the literature by applying robust statistical validation.

The objectives were achieved. The hybrid system was implemented, and the proposed system was compared to a method from the literature.

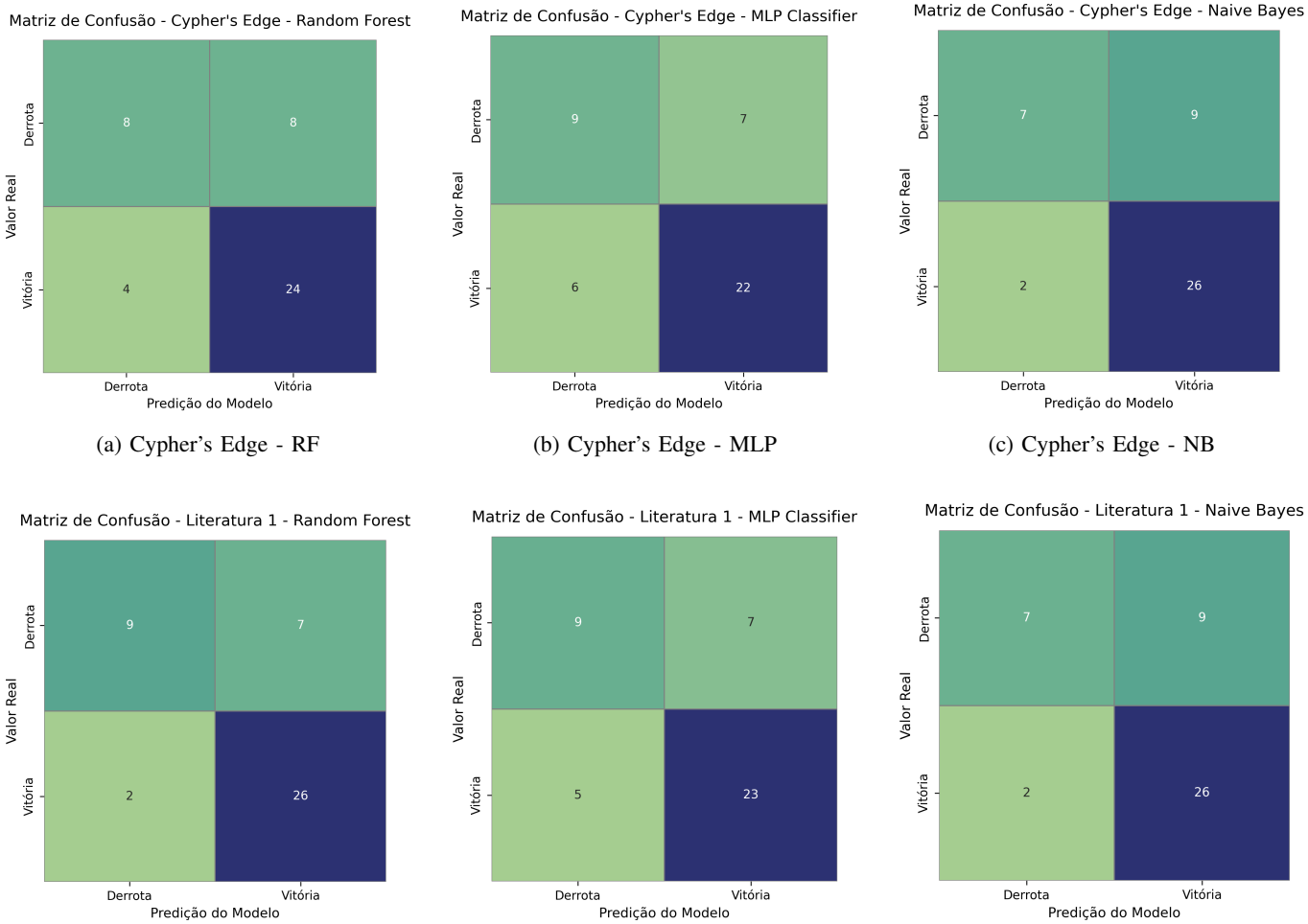
The experiments, statistically validated by the Nemenyi test (Fig. 9), demonstrated that **the three classifiers (RF, MLP, NB) are statistically equivalent** for this problem, although Random Forest achieved a slightly better (but not significant) numerical ranking. This finding suggests that simpler models (like Naive Bayes) may be sufficient, challenging the pursuit of complex models.

The relevance analysis (Fig. 5) identified **turretTakedowns** and **goldEarned** as the attributes with the greatest impact, a practical result that the system can provide to the player.

As a limitation, the "Cypher's Edge"dataset (N=48) is small, which prevents the generalization of results. For future work, a massive expansion of the database is suggested to validate these findings and include more complex metrics, such as "Victory Contribution"(VC) [4], and other pre-game data [5].

Referências

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



(a) Cypher's Edge - RF (b) Cypher's Edge - MLP (c) Cypher's Edge - NB

(d) Literatura 1 - RF (e) Literatura 1 - MLP (f) Literatura 1 - NB

Figura 7: Confusion matrices for all evaluated classifiers.

[2] F. Pedregosa *et al*., "Scikit-learn: Machine learning in Python," *Journal of machine learning research*, vol. 12, pp. 2825–2830, 2011.

[3] S. Pan, T. Kim, e J. Smith, "Applications of Linear and Ensemble-Based Machine Learning for Predicting Winning Teams in League of Legends," *IEEE Transactions on Games*, vol. 11, no. 4, pp. 340–347, Dez. 2019.

[4] Y. Kim, H. Lee, e S. Lee, "A role-sensitive framework for evaluating player contributions and predicting match outcomes in League of Legends," in *Proc. ACM SIGKDD Conf. on Knowledge Discovery and Data Mining*, 2018, pp. 1655–1664.

[5] J. Doe e A. Silva, "External factors in MOBA performance: Analyzing toxicity and communication," *Journal of Online Gaming Research*, vol. 5, no. 1, pp. 45–59, 2020.