Development of an Automated Machine Learning Solution Integrable With Multiple Virtual Learning Environments

Raniel Gomes da Silva[1], Vitória Maria Pena Mendes[1], Rodrigo Lins Rodrigues[2], and Alexandre Magno Andrade Maciel[1]

[1]Universidade de Pernambuco
[2]Universidade Federal Rural de Pernambuco

Abstract

In the last decade, a large volume of data has emerged from the massive use of Virtual Learning Environments (VLE). The information contained in these data has enabled the evolution of Educational Data Mining (EDM), whose objective is to apply Machine Learning (ML) in educational contexts. However, building accurate and robust ML models requires, in most cases, advanced knowledge in data science. To solve such problems, Automated Machine Learning techniques have been studied, to simplify the repetitive processes of Data Mining. To validate the solution, the database of the Núcleo de Educação a Distância da Universidade de Pernambuco was used. In comparison with the classic EDM approaches, the applied technique showed a superior result, obtaining an accuracy of 89% in the student performance classification process. This solution is called the Framework de Mineração de Dados Educacionais (FMDEV), whose objective is to allow users to validate and make available ML baselines with greater productivity. The results of the experts' opinions prove that the FMDEV can contribute to the construction of better models of ML.

1 Introduction

In the last decade, the adoption of distance learning tools has grown exponentially. Consequently, a large volume of data has emerged from the massive use of Virtual Learning Environments (VLE) [1]. For knowledge extraction from these data, it is necessary to carry out a series of data mining processes [2].

Educational Data Mining (EDM) techniques have been adopted frequently, as an alternative for extracting knowledge from data obtained from VLE [3]. The EDM process is conceptualized as a paradigm for building models, tasks, methods, and algorithms from educational databases. Within this context, the use of supervised learning is quite common to solve tasks such as analyzing student performance or predicting evasion risk. [4].

Machine Learning (ML) application is important for data scientists, tutors, and teachers. However, given the complexity of educational problems, the model building requires advanced knowledge in data science [5]. In addition, other factors such as the development time in model building, the definition of input parameters, and the selection of the best algorithm, make it impossible for the use of ML to be democratized for non-technical users. [6].

To solve such problems, Automated Machine Learning (AutoML) techniques have been studied, with the objective of simplifying repetitive Data Mining processes, which do not require domain knowledge in most cases [7]. Bayesian Optimization (BO) and Evolutionary Algorithm (EA) techniques have been applied in the categories of Automated Engineering of Features (AutoFE) and in the Automation of Models with Hyperparameter Learning (AutoMHL) [8].

Some technologies simplify such steps, such as TPOT [9, 10], AutoKeras [11] and Auto-Weka [12], however, it is necessary to have a minimum of knowledge in data science [13]. Tools like the one by Fusijawa et al., present a closer path for a non-technical user, however, the developed features require knowledge in Structured Query Language (SQL) and also ML, as the user needs to define an algorithm for training [14].

Thus, the justification for this work lies in the need to develop an AutoML solution that favors users in the extraction of knowledge from EDM, even with little experience in ML techniques. Considering the various possibilities of applications of this solution, this research focused on EDM, with the integrability of multiple VLE, to obtain evidence in the context of a course, discipline, period, or class.

The general objective of this article is to develop an Automated Machine Learning solution that can be integrated into Virtual Learning Environments or data visualization tools, based on the application of models generated through Genetic Algorithm techniques. To achieve the general objective, the following specific objectives were established:

DOI reference number: 10.18293/SEKE2021-068

Identify the essential functionalities of an Automatic Machine Learning solution for use in Educational Data Mining; Implement an application with the proposed solution and evaluate the implemented solution from experiments with the database of the Núcleo de Educação a Distância da Universidade de Pernambuco (NEAD) and measure the quality of the framework from integration tests.

2 Background and Related Work

2.1 Machine Learning

Machine Learning has as its definition the application of computational methods that obtain expertise, with the objective of improving performance or applying partial predictions in a given context [15]. In practice, prediction algorithms are built with high robustness, which depends on a sample necessary for the algorithm to learn a family of concepts [15]. Contextualizing this task for the CRISP-DM, it is possible to fit the ML in the pre-processing, modeling, and evaluation steps.

As for the types of ML algorithms, are present the supervised learning, unsupervised learning, semi-supervised and reinforcement learning [16]. The definition of the correct type for each context may depend on some criteria, such as (I) The need for human supervision as to the expected output; (II) Speed of relearning as new input data is available; (III) The identification of new patterns based on trained characteristics, or simply, classifying them considering a data entry [16].

2.2 Automated Machine Learning

Automated Machine Learning (AutoML) is an abstraction for ML in which it proposes to optimize productivity in the pre-processing and modeling steps. Steps such as selecting features, defining the best algorithm and configuring hyperparameters, are built automatically and without human intervention [17]. Another value delivered by the AutoML approach is to prevent the data scientist from wasting development time on repetitive trial and error tasks [17].

There are categories in AutoML with a greater focus on the selection of features (AutoFE) and others with a greater emphasis on the definition step of model optimization and hyperparameters (AutoMHL), finally cases more focused on Deep Learning (AutoDL). Among the techniques most applied in AutoML, there are Bayesian Optimization (BO) and Evolutionary Algorithm (EA) [8]. There are also applications based on techniques with Reinforcement Learning and Gradient-based, but they are still very incipient [8].

Figure 1 defines a conventional ML flow. According to the diagram above, the use of AutoML seeks to optimize the two major areas in evidence. From the point of view of features engineering, three steps seek to be optimized. In the data cleaning step, imputation techniques and attribute normalization are applied. When generating features, new metadata is created from existing ones. In the selection of features, dimensionality reduction techniques (VarianceThreshold, SelectKBest), to optimize the database. From the modeling point of view, BO or EA techniques are applied to identify the best algorithm and hyperparameters (Figure 1).

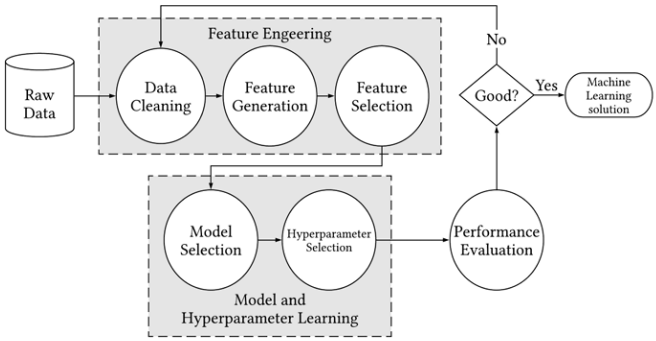

Figure 1: 2019 Automated Machine Learning default architecture [8]

3 Proposed Solution

The proposed solution is called the Framework de Mineração de Dados Educacionais (FMDEV). FMDEV applies the CRISP-DM [18] methodology to the solution steps. On the construction of the solution, Requirements Engineering [19] and Lean Inception [20, 21, 22] techniques were applied to define the necessary functionalities for FMDEV.

FMDEV is able to assist in data pre-processing, training, validation, and availability. The steps of understanding the business and understanding the data are not the responsibility of FMDEV. FMDEV makes the models available in REST format so that data visualization tools and Virtual Learning Environments are able to consume the endpoints generated by FMDEV.

Before the actual implementation process, a navigable prototype was built using the tool Figma [23]. The mainstream of Framework de Mineração de Dados Educacionais (FMDEV) is divided into four screens: (I) Data sources; (II) Selection of indicators; (III) Data pre-processing and (IV) Training. For the trained and saved models, a screen was created separately from the main flow.

4 Results

The assessment of the environment proposed in this work sought to validate the FMDEV under four sets of experiments: the first analyzed the use of it as a tool to assist in the generation of educational data mining models; the second assessed the holistic functioning of the environment based on integration tests; the third, carried out an opinion

Course	ROC AUC	Recall	Precision	F1 Score
Administration	96,42%	80,33%	86,80%	83,42%
Biology	93,34%	80,32 %	87,34%	83,67%
Literature	88,77%	77,39%	82,94%	80,06%
Pedagogy	93,16%	78,25%	87,58%	82,65%

Table 1: Bayesian Optimization Technique Results.

survey with experts in the field of data science, addressing the contributions that the tool provides to the development of machine learning models; and the last, evaluated the usability of FMDEV in order to verify how the environment can establish a better user experience to non-technical users. The following subsection will present only the experiment of educational data mining models due to the limit of this work.

4.1 Generation of Data Mining Models

To validate the model generated by FMDEV, an experiment was carried out with the database provided by NEAD. The base is a backup of the MySQL Relational Database Management System (RDBMS) of the Moodle Learning Management System. This database contains 30,218 instances, which includes courses in Administration, Biology, Literature, and Pedagogy, referring to the years 2010 to 2016. The construction of these variables occurred from a set of SQL queries. The result of these queries allowed 33 variables relevant to the students' characteristics to be created.

From this, four sub-bases were created, each one related to a course (Administration, Biology, Literature, and Pedagogy). As for the number of instances in each course, it is divided as follows: Administration (2892), Biology (6526), Literature (6297), and Pedagogy (14502).

Regarding the problem to be solved, it is about the analysis of student performance from the way it interacts with Moodle. The supervised models built used the DESEMPENHO_BINARIO variable as the target variable, whose classes are 0 and 1. Class 0 indicates that the student failed and class 1 indicates that the student passed.

Each experiment made use of the AutoML techniques presented in this work. With this, eight approaches were applied in total (4 courses * 2 techniques). Both techniques were performed with five epochs. In the case of the GA technique, epochs are called generations. The input parameters of GA also depend on the population size (configured with 100), mutation rate (configured with 0.9), and crossing rate (configured with 0.1).

For each scenario, the base was divided into 70% training and 30% tests. As for the evaluation metrics, ROC curve (AUC), recall, precision, and F1 Score are present. All procedures were performed on a Linux server (Intel Core i7 2.2 GHz; 4 Cores; 16 GB of RAM). According to Table 1, the first round of tests included the BO technique.

AutoKeras library was applied to assist the execution of the AutoML technique for Bayesian Optimization [24]. It is possible to notice that the Administration course obtained the best performance in the tests (89.36%). The Literature course had the lowest accuracy (84.23%). In the 2 table, the results with the GA technique and the due considerations regarding the two techniques will be presented.

Course	ROC AUC	Recall	Precision	F1 Score
Administration	96,62%	84,51%	91,92%	88,04%
Biology	95,93%	84,37%	90,95%	87,47%
Literature	94,23%	86,87%	87,16%	87,01%
Pedagogy	96,11%	84,92%	88,99%	86,90%

Table 2: Genetic Algorithm Technique Results.

TPOT library was applied to assist the execution of the AutoML technique for Genetic Algorithm [25]. Specifically for this technology, it is possible to create a parallelism rule, in which all colors of the experiment machine can be used [26].

As for the models optimized for each experiment with GA, the following algorithms were obtained: XGBoost (Administration and Biology), Random Forest (Literature) and Extra Trees (Pedagogy) [27, 28, 29]. It is interesting to note that the administration and biology courses obtained the same algorithm, however, their hyperparameters, defined from the multiple generations in the AutoML technique, obtained completely different configurations.

When comparing the metrics of GA and BO, it is noticeable that the GA excels in all cases. From an average among all courses, the accuracy of GA is 2.52% higher; AUC at 2.79%, recall at 6.08%, precision at 3.57% and F1 Score at 4.90 %. Of the four courses evaluated, the greatest discrepancy in techniques is presented in the Literature course. The F1 Score generated by the GA technique for this course, is 6.94 % higher. Given this, it is important to note several contributions with the use of the GA technique in AutoML: superior results in relation to the BO technique and explanability of the models and the optimized hyperparameters.

5 Conclusion

This work proposed the development of an Automated Machine Learning solution for EDM. For this, a framework was developed capable of using Moodle data sources, as well as CSV files that can be imported directly into FMDEV. The framework allows you to create, manage, and consume supervised classification models in a simple way for non-technical users and productive for technical users or with little expertise in data science.

Regarding the machine learning area, contributions were reported from the use of the Genetic Algorithm techniques, in comparison with the Bayesian Optimization technique. For the case study applied with the NEAD database, it was possible to perceive that the results presented from the GA technique, was superior in all scenarios, in comparison with

the BO technique. In addition, the GA technique showed superior results compared to conventional data mining techniques, considering the case study addressed.

Regarding the software engineering area, integration tests were developed for all available endpoints. Such tests ensured that the FMDEV presents consistency and conformity based on the functional and non-functional requirements presented. In addition, these tests will be useful as a software quality assurance strategy.

As a way of ensuring more reliability in this research, an expert opinion was carried out to assess the conformity of the steps developed in the FMDEV. The experts' feedback corroborated that FMDEV is able to simplify the process of mining educational data, as well as promoting productivity in the use of automated machine learning. The fact that FMDEV is able to abstract the complexity of Machine Learning algorithms, shorten the development time of models and remove the difficulty in defining the parameters of the algorithms, will enable a great differential in the processes of EDM.

Acknowledgement

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - finance code 001.

References

[1] C. Patel, M. Gadhavi, and A. Patel, "A survey paper on e-learning based learning management systems (lms)," *International Journal of Scientific & Engineering Research*, vol. 4, no. 6, pp. 171–177, 2013.

[2] C. Romero and S. Ventura, "Educational data mining: A review of the state of the art," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 40, no. 6, pp. 601–618, Nov. 2010. [Online]. Available: https://doi.org/10.1109/tsmcc.2010.2053532

[3] G. Mahajan and B. Saini, "Educational data mining: A state-of-the-art survey on tools and techniques used in edm," 2020.

[4] A. Peña-Ayala, "Educational data mining: A survey and a data mining-based analysis of recent works," *Expert Systems with Applications*, vol. 41, no. 4, pp. 1432–1462, Mar. 2014. [Online]. Available: https://doi.org/10.1016/j.eswa.2013.08.042

[5] S. Viaene, "Data scientists aren't domain experts," *IT Professional*, vol. 15, no. 6, pp. 12–17, Nov. 2013. [Online]. Available: https://doi.org/10.1109/mitp.2013.93

[6] M. Tsiakmaki, G. Kostopoulos, S. Kotsiantis, and O. Ragos, "Implementing AutoML in educational data mining for prediction tasks," *Applied Sciences*, vol. 10, no. 1, p. 90, Dec. 2019. [Online]. Available: https://doi.org/10.3390/app10010090

[7] R. Elshawi, M. Maher, and S. Sakr, "Automated machine learning: State-of-the-art and open challenges," 2019.

[8] Y.-W. Chen, Q. Song, and X. Hu, "Techniques for automated machine learning," 2019.

[9] R. S. Olson, N. Bartley, R. J. Urbanowicz, and J. H. Moore, "Evaluation of a tree-based pipeline optimization tool for automating data science," in *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference - GECCO '16*. ACM Press, 2016. [Online]. Available: https://doi.org/10.1145/2908812.2908918

[10] R. S. Olson and J. H. Moore, "TPOT: A tree-based pipeline optimization tool for automating machine learning," in *Automated Machine Learning*. Springer International Publishing, 2019, pp. 151–160.

[11] H. Jin, Q. Song, and X. Hu, "Auto-keras: An efficient neural architecture search system," 2018.

[12] L. Kotthoff, C. Thornton, H. H. Hoos, F. Hutter, and K. Leyton-Brown, "Auto-weka 2.0: Automatic model selection and hyperparameter optimization in weka," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 826–830, 2017.

[13] A. Truong, A. Walters, J. Goodsitt, K. Hines, C. B. Bruss, and R. Farivar, "Towards automated machine learning: Evaluation and comparison of AutoML approaches and tools," in *2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)*. IEEE, Nov. 2019. [Online]. Available: https://doi.org/10.1109/ictai.2019.00209

[14] I. Y. Fujisawa and A. M. A. Maciel, "Desenvolvimento de um framework integrador de mineração de dados educacionais," *Revista de Engenharia e Pesquisa Aplicada*, vol. 3, no. 3, Sep. 2018. [Online]. Available: https://doi.org/10.25286/repa.v3i3.977

[15] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of machine learning*. MIT press, 2018.

[16] A. Géron, *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. O'Reilly Media, 2019.

[17] B. Chen, H. Wu, W. Mo, I. Chattopadhyay, and H. Lipson, "Autostacker: A compositional evolutionary learning system," 2018.

[18] R. Wirth and J. Hipp, "Crisp-dm: Towards a standard process model for data mining," in *Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining*. Springer-Verlag London, UK, 2000, pp. 29–39.

[19] B. H. Cheng and J. M. Atlee, "Research directions in requirements engineering," in *Future of Software Engineering (FOSE '07)*. IEEE, May 2007. [Online]. Available: https://doi.org/10.1109/fose.2007.17

[20] L. Wilson, *How to implement lean manufacturing*. McGraw-Hill New York, 2010.

[21] J. N. R. Boeira, "First steps with lean," in *Lean Game Development*. Springer, 2017, pp. 9–21.

[22] P. Caroli, "Lean inception: How to align people and build the right product," *Rio de Janeiro, Brasil: Caroli Editora*, 2018.

[23] "Figma: the collaborative interface design tool." https://www.figma.com/, (Accessed on 08/17/2020).

[24] "Autokeras," https://autokeras.com/, (Accessed on 08/12/2020).

[25] "Epistasislab/tpot: A python automated machine learning tool that optimizes machine learning pipelines using genetic programming." https://github.com/EpistasisLab/tpot, (Accessed on 08/19/2020).

[26] "Automate machine learning with tpot — dask examples documentation," https://examples.dask.org/machine-learning/tpot.html, (Accessed on 08/19/2020).

[27] T. Chen, T. He, M. Benesty, V. Khotilovich, and Y. Tang, "Xgboost: extreme gradient boosting," *R package version 0.4-2*, pp. 1–4, 2015.

[28] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[29] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Machine learning*, vol. 63, no. 1, pp. 3–42, 2006.