Higher Education Studies; Vol. 10, No. 4; 2020
ISSN 1925-4741 E-ISSN 1925-475X
Published by Canadian Center of Science and Education

A Machine Learning-Based Computational System Proposal Aiming at Higher Education Dropout Prediction

Maria do Carmo Nicoletti[1, 2] & Osvaldo Luiz de Oliveira[1]

[1] Centro Universitário Campo Limpo Paulista (UNIFACCAMP), C. L. Paulista, SP, Brazil

[2] Computer Science Dept., Universidade Federal de S. Carlos, S. Carlos, SP, Brazil

Correspondence: Maria do Carmo Nicoletti, UNIFACCAMP, Rua Guatemala, 167, Jardim América, 13231-230 Campo Limpo Paulista, SP, Brazil. Tel: 55-114-812-9400. E-mail: carmo@cc.faccamp.br

Received: August 11, 2020 Accepted: September 3, 2020 Online Published: September 10, 2020

doi:10.5539/hes.v10n4p12 URL: https://doi.org/10.5539/hes.v10n4p12

Abstract

In the literature related to higher education, the concept of dropout has been approached from several perspectives and, over the years, its definition has been influenced by the use of diversified semantic interpretations. In a general higher education environment dropout can be broadly characterized as the act of a student engaged in a course leaving the educational institution without finishing the course. This paper describes the proposal of the architecture of a computational system, PDE (Predicting Dropout Events), based on machine learning (ML) algorithms and specifically designed for predicting dropout events in a higher level educational environment. PDE's main subsystem implements a group of instance-based learning (IBL) algorithms which, taking into account a particular university-course environment, and based on log files containing descriptions of previous dropouts events, is capable to predict when a student already engaged in the course, is prone to dropout, so preventive measures could be quickly implemented.

Keywords: machine learning based systems, relevant feature selection, instance-based learning, dropout at undergraduate level

1. Introduction

In educational environments related to traditional learning or e-learning, the dropout problem has been recurrent for many years already and, so far, it has shown to be a difficult problem to be solved, even when both, the administrative staff and the educational staff are heavily engaged in its solution. The difficulties to successfully address the problem are mainly due to its many dimensions, such as: the need for correctly detecting the relevant variables involved in the process and how they relate to each other; the volatility of many of the variables involved; the influences of the social environment that embeds the educational environment; the influences of the internal social environment, that is an inherent part of the educational system, etc.

The most popular and influential among the earlier proposals of higher education dropout models are those described by Bean (1980, 1985), by Bean and Metzner (1985), by Pascarella (1980), by Pascarella and Chapman (1980), by Tinto and Cullen (1973) and by Tinto (1975, 1997). In the literature related to higher education, however, the process of dropout has been approached from several slightly different perspectives and, over the years, the definition of dropout has been adapted to conform to the growing number of influential new factors and changes occurring in higher level educational environments.

As a consequence of the constant growing dynamism of higher educational environments, dropout processes have also acquired a highly dynamic structural nature, in opposition to the static structural nature assumed by the earlier dropout modelling proposals, such as those characterized as earlier proposals. Some of the earlier models, however, have underdone upgrading trying to to keep pace with changes and have been used by the higher education community for both, experiments as well as guidelines for analyzing and avoiding dropout events, such as those described by Belloc and Maruotti and Petrella (2010), by Chrysikos and Ahmed and Ward (2017), by Dekker and Pechenizkiy and Vleeshouwers (2009), by Durso and Cunha (2018), by Murray (2014), by Smith and Naylor (2001), by Tinto (1997), by Vergidis and Panagiotakopoulos (2002), by Willging and Johnson (2004) and by Kerby (2015).

The dropout process's dynamic is particularly related to both, the number and the nature of the subprocesses that it encompasses, as well as the number and type of the variables involved in the process. The interrelationships between subprocesses and the variables they share also play an important role in the whole dropout process. Usually variables, subprocesses and interrelationships are highly volatile over time, reflecting the dynamics of current educational environments.

In such scenario, the only way of monitoring the dynamics of dropout processes, aiming at preventing them, is through a computational system capable to detect patterns of student's behavior that may lead to a dropout decision. The system should be reactive enough to detect 'symptoms' of eventual dropout events, as earlier as possible, within a frame of time suitable for accommodating the implementation of preventive measures.

This paper describes the proposal of one of such systems and for that, it is organized as follows. In Section 2 some of the most influential dropout models available in the literature are briefly reviewed, with the aim to provide the theoretical grounds that supported part of the proposal described in this paper. Considering the main focus of this paper is on the proposal of a machine-learning (ML) based computational system for dropout prediction, Section 3 presents the main technicalities related to a family of five algorithms, characterized as Instance-based learning (IBL), as proposed by Aha, Kibler, & Albert (1991); Aha (2013) and two other IBL algorithms, by Gates (1972) and by Hart (1968), which composes the core of such system. The section also highlights algorithms that can contribute for preparing the raw educational data available, by removing inconsistences, noise and eventually, irrelevant attributes, a phase known as data pre-processing. In Section 4 the specification of the dropout prediction system PDE is presented and discussed, with the suggestion of possible software architecture for implementing it. Section 5 summarizes the main ideas described in the paper and the motivations for the computational system proposed, stressing the need for a careful choice of the set of variables that support the decision of dropout or persistence, the relevance of having the collected data used for ML learning, free from the common problems found in raw data, as well as the importance of having a clear definition of all the involved processes based on these variables.

2. Earlier Modelling of Higher Education Dropout

The model for university dropout process, known as Model of the Dropout Process (MDP), was proposed by Spady (1970, 1971). Besides being considered one of the first of such models, its relevance is also credited to the fact that is considered to be one of the first dropout's modelling that took into account sociological variables, believed to be intrinsically involved in the process.

The MDP models the dropout event as a temporal process based on information from two subjacent intertwined subprocesses, the academic and the social, since both co-exist in higher-level institutions. As pointed out by Nicoletti (2019), in his both papers Spady presents a very detailed discussion about the variety of issues as well as the many variables involved in dropout processes, and provides, as support for his analysis and comments, results found in a large number of empirical and theoretical works in the literature.

The MDP model can be approached as a temporal sequence of nine interrelated modules. When the final module in the sequence is reached~~,~~ the process ends, meaning that the dropout decision had been made by the student. Spady's model represents the decision for dropout as an iterative process, throughout the period the student is engaged in the educational institution. The first module in the sequence, the Family Background, provides information about the student's background. The two following modules are the Academic Potential and the Normative Congruence. Usually the Academic Potential module gathers data related to well-known tangible academic variables, such as previous (to entering university) academic marks and honors.

Spady (1971) defines Normative Congruence as "... the concept of normative congruence refers to the general degree of compatibility between the dispositions, interests, attitudes, and expectations of the student and the set of behaviors, expectations, and demands to which he may be exposed as the result of interaction with a variety of individuals in the college environment. To the extent that these expectations and influences are highly consistent within a given college context, it is presumed that students whose attributes enable them to accommodate themselves readily to these influences will experience less strain in their general interaction with others, be they fellow students, faculty members or administrators." As commented by Spady (1970, p. 78), a problematic aspect of his model is related to the meaning and operationalization of the Normative Congruence module, considering the critical role it plays in the model. As Spady points out "It represents not only all of the student goals, orientations, interests, and personality dispositions discussed earlier, but the consequences of the interaction between these attributes and various subsystems of the college environment as well."

The concept of Normative Congruence is specific to the MDP and involves variables difficult to be characterized and, consequently, to be measured (Nicoletti, 2019). Both, the Academic Potential and the Normative Congruence modules influence some of the modules as follows. The Academic Potential influences the Grade

Performance and the Intellectual Development modules, while the Normative Congruence, besides influencing the Academic Potential and the Grade Performance, also influences two others in the sequence, the Friendship Support as well as the Social Integration modules, respectively.

The Friendship Support module also influences the Grade Performance and the Intellectual Development modules as well as the Social Integration module. Besides being influenced by information provided by the Friendship Support module, the Social Integration module is influenced by three other previous modules namely, Normative Congruence, Grade Performance and Intellectual Development. By being fed with information from the four modules and by evaluating the degree of the student's social integration into both subsystems i.e., the social and the academic, the Social Integration module stresses the important role played by the student's social integration, in the MDP model. According to Spady the Satisfaction module's output can be considered the outcome of the MDP model or, then, as information to be passed on to the Institutional Commitment module, which combines both, student's satisfaction with the course and his/her commitment to the educational institution, to support his/her decision for dropout or not.

The way the dropout process has been diagrammatically represented, when the flow of information reaches the Institutional Commitment module, if the dropout decision has not been reached yet, the process iterates back to the Normative Congruence module and repeats itself from there. Aljohani (2016) commented that only after the release of the Spady's model, the research related to university dropout problem started to take into account the student-institution interaction as one of the many relevant information involved in the process.

The work by Tinto and Cullen (1973) describes the results of a survey commissioned by the U.S. Office of Education, by presenting a theoretical synthesis of research works on dropouts in higher education at the time. The third part the survey describes the proposal of a basic theoretical model aimed at explaining dropout as an interactive process involving both, student and educational institution.

In a general approach the Tinto's original model resembles the Spady's model, taking into account that both focus on similar groups of variables that are considered relevant to the dropout decision. Since its proposal the Tinto's model has become very popular and has been employed in several works with the intent of detecting dropout events before they happen. In (Tinto, 1975) the original Tinto's model was reviewed and modified, but still maintaining the original sequence of modules, although with smaller granularity and also, introducing External Communities modules, as the providers of variable values related to external influences on dropout processes. In (Tinto, 1997) new refinements were introduced and particularly, the temporal aspect of the model was emphasized by the presence of a time line.

As pointed out by Nicoletti (2019), "… the Goal commitment (module in Tinto's model) is a very complex issue and, definitely, is a multidimensional variable, composed by several other variables, each possibly having a multitude of values. Possibly, a few existing variables it involves can even not be known to be relevant to the process. As it can be inferred from the model, independently of the process followed to assign to Goal commitment a value, the mechanism in charge of calculating this value considers only three groups of variables i.e., those in module Family Background, those in module Individual Attributes and those in module Pre-College Schooling." Tinto's model is still largely used by educational institutions as a way of organizing and directing both, the study and the analysis of the dropout problem in situ, aiming at solving it.

Pascarella (1980) and Pascarella & Chapman (1980) propose a theoretical dropout model where student-faculty informal contacts play a relevant role on the persistence/dropout process. The model's structure has five modules known as: Student Background Characteristics, Institutional Factors, Informal Contact with Faculty, Other College Experiences and Educational Outcomes. Some of the interactions between modules are bi-directional, such as those between the Institutional Factors and the Student Background Characteristics modules. Particularly three modules, Informal Contact with Faculty, Other College Experiences and Educational Outcomes, provide each other information in a circular way, via three bi-directional connections. As pointed out by (Nicoletti (2019), "in spite of the emphasis on the student-faculty informal contact, the model also takes into consideration the vast amount of information that lies underneath the academic life of a student, as some of the previous models do (e.g. Spady's model and Tinto's model)."

A pragmatic aspect attached to the Pascarella's model refers to the group of variables associated to each of the five modules of the model, as shown in Table 1. As pointed out by Nicoletti (2019), "A relevant and difficult aspect to consider in relation to the Pascarella's model (and, in an extent, to several theoretical dropout models) is how to determine at which level the variables in Table 1 contribute to the decision of dropout and, also, how to define them so they can be tangible variables and, still, represent the semantics of their names." Also, several of them can be approached as a composition of single variables.

Table 1. Variables involved in the five main modules of Pascarella ́s model.

Module	Variables
Student Backgroud Charactreristics	• Family Background
	• Aptitudes
	• Aspirations
	• Personality Orientations, Goals, Values, Interests
	• Secondary School Achievements & Experiences
	• Expectations of College
	• Openness to Change
Institutional Factor	• Faculty Culture (e.g. professional interests, values, orientations)
	• Organizational Structure
	• Institutional Image
	•Administrative Policies & Decisions
	• Institutional Size
	• Admissions Standards
	• Academic Standards
Informal Contact with Faculty	• Context
	• Exposure
	• Focus
	• Impact
Other College Experiences	• Peer Culture
	• Classroom
	• Extracurricular
	• Leisure Activities
Educational Outcomes	• Academic Performance
	• Intellectual Development
	• Personal Development
	• Educational/Career Aspirations
	• College Satisfaction
	• Institutional Integration

A literature review in the research area related to modeling the dropout problem in undergraduate courses, as commented in (Nicoletti 2019), can be systematically approached considering the high number of research works in this particular area, classified into four large groups, depending on their main goal being related with:

1) empirical validation of some of the earlier dropout models briefly reviewed in Section 2, such as those described by Chrysikos and Ahmed and Ward (2017) and by Durso and Cunha (2018). In this group of work can also be included case studies supported by earlier models as well as ad hoc models such as works by Dekker and Pechenizkiy and Vleeshouwers (2009), by Belloc and Maruotti and Petrella (2010), by Gordon (2016), by Wiseman and Gonzales and Salyer (2004), by Paura and Arhipova (2014) and by Giannakos et al. (2017).

2) new proposals of dropout models that combine early models with some ad hoc relevant characteristics of particular learning environments, such as the model by Kerby (2015), that combines three earlier models, and the model proposed by Rovai (2003).

3) detecting variables and factors that are relevant for predicting dropout/persevere, taking into consideration the available data (Giannakos et al., 2017; Murray, 2014; Pidgeon, Rowe, Stapleton, Magyar, & Lo, 2014; Willging & Johnson, 2004; Xenos, Pierrakeas, & Pintelas, 2002; Nicoletti, Reali, Dias, & Abib, 2012).

4) revision works (Aljohani, 2016; Demetriou & Schmitz-Seiborski, 2011).

Models of student persistence (Ethington, 1990) or student dropout processes have been a recurrent issue in the literature related to higher-level educational institutions. Besides the briefly described earlier models, new proposals can be found in the literature, which usually combine aspects of the previous models, such as those described in (Kerby, 2015) and (Rovai, 2003). For the proposal of the PDE’s system, several issues related with the previously reviewed models have been considered, as well as algorithms from two important subareas of Artificial Intelligence namely, Machine Learning (ML) and Feature Selection (FS), which are the focus of the following two sections, respectively.

3. Instance-Based Learning Algorithms and Preprocessing Educational Data

The vast majority of ML algorithms adopt the inductive approach i.e., they implement a process that, based on a given set of data instances (training set) induces, via a generalization process, a description (such as a neural network, a decision tree, a set of decision rules, a set of data points, flow graphs, etc.) of the concept(s) embedded in the data, which is known as the *expression of the concept*.

In what follows relevant concepts and algorithms employed in the proposal of the PDE's computational system are briefly reviewed, with the intent to provide the necessary grounds for the understanding of the system and ease its implementation. Section 3.1 has its focus on instance-based algorithms and presents a brief description of several IBL algorithms which can be considered to be part of the PDE's learning module, individually or implemented as an ensemble of IBL-based classifiers. Section 3.2 has its focus on preprocessing the educational data available aiming to remove eventual problems commonly found in raw data.

3.1 Instance-based Learning

Instances in training sets are usually described by vectors of attribute values and, depending on the situation, an associated class indicating the concept the instance represents. The class of each training instance is usually determined by a human expert. When the class is used by the inductive process implemented by a ML algorithm, the algorithm is a supervised algorithm and, usually, the expression induced by the algorithm is referred to as a classifier.

ML supervised algorithms usually have two phases: (1) the training phase where, given a training set of instances, the algorithm generalizes it into a general expression and (2) the testing (classification) phase, where the induced expression of the concept is used for classifying new data instances of unknown class.

Supervised algorithms characterized as Instance-Based Learning (IBL) (Cover & Hart, 1967; Hart, 1968; Gates, 1972; Aha, Kibler, & Albert, 1991; Aha, 2013) are of particular interest in this work, since they can be easily implemented and frequently have good predicting results. Usually in their training phase IBL algorithms simply store the training instances. The generalization process which commonly happens in the training-phase for most ML algorithms is postponed until a new instance, of unknown class, needs to be classified. The concept representation adopted by IBL algorithms i.e., that of assuming the given set of data instances as the expression of the concept, is the simplest form of representation, since the learning process consists simply in "memorizing" the training set.

An advantage of this type of learning is that, instead of implementing a commonly elaborated mechanism for generalizing the concept that takes into account the entire training set available, it conducts the generalization process in the classification phase, by estimating the concept locally, for each new instance to be classified. One of the disadvantages of IBL algorithms is the computational cost of classifying new instances, when the training set is bulky and, also, data instances are described by a high number of attributes, since the entire processing required takes place in the classification phase and demands the calculation of the proximity between the new instance and each of the stored instances.

Usually for calculating the proximity between two instances in a *d*-dimensional Euclidean space, the Euclidean distance formula is employed. As pointed out by Marshland (2009), for calculating the Euclidean distance between two instances in a *d*-dimensional Euclidean space, *d* subtractions and *d* exponentiations with exponent 2 must be performed. The calculation of the square root in the Euclidean distance formula can be avoided, since the goal is to measure proximity between the two instances and not the exact distance between them. Such calculations are of order O(N), where N represents the number of stored instances.

The Nearest Neighbor (NN) (Cover & Hart, 1967) is a successful IBL algorithm which has also inspired the proposal of many others. The NN usually employs a distance function to determine the highest degree of similarity between the new instance to be classified and each one of the stored instances, assuming similarity as the inverse of the distance between the two instances. When such instance is selected, the new instance inherits its class. Figure 1 shows the pseudocode of the NN algorithm based on the description found in (Gates, 1972), where a set of N data instances, described by M-dimensional vectors, each of them associated with one out of S classes, is the input to the algorithm. The NN can be extended into a version known as k-NN which, instead of searching for the closest instance to the instance of unknown class, searches for the k closest instances.

Algorithms IB1, IB2, IB3, IB4 and IB5 (Aha, Kibler, & Albert, 1991; Aha, 2013) were proposed with the intent of experiencing with the limits of IBL algorithms. IB1 is a version of the NN that stores the whole training set in an incremental way and, also, keeps performance records associated with each stored training instance. IB2 focuses on reducing the number of stored instances while tries to maintain the predictive power unchanged; the

IB2 can be considered a variant of the Condensed Nearest Neighbor (CNN), proposed by Hart (1968). IB3 is a variant of the IB1 proposed with the intent to explore the effects of noisy data on classification results and, also, as an attempt to reduce the sensitivity of the IB2 algorithm to noisy data. IB4 is a variant of IB3 that seeks to circumvent its sensitivity to irrelevant attributes in the description of data instances, and the IB5 is a variant that tries to explore the sensitivity of an IBL algorithm to the introduction of new attributes into the instances' description.

```
Training algorithm:
• store training set with N training instances,
    $TS_{NN} = \{(x_1, \theta_1), (x_2, \theta_2), ..., (x_N, \theta_N)\}$,
    where:
      (1) $x_i$ $(1 \le i \le N)$ is a M dimensional vector of attribute values:
              $x_{i} = (x_{i_1}, x_{i_2}, ..., x_{i_M})$
      (2) $\theta_i \in \{1, 2, ..., S\}$, is the correct class of $x_i$ $(1 \le i \le N)$.

Classification algorithm:
    • given an instance $x_q$ to be classified, the decision rule implemented
      by the algorithm decides that $x_q$ has class $\theta_j$ if
              $d(x, x_j) \le d(x, x_i)$          $1 \le i \le N$
      where d is a M-dimensional distance metric.
```

Figure 1. High level pseudocode of the Nearest Neighbor (NN) (Cover & Hart, 1967)

The CNN algorithm was proposed by Hart (1968) with two goals: to reduce the set of instances to be stored such that the remaining instances would still be able to maintain a good predictability for the new instances to be classified. The CNN still classifies a new instance by taking into account the class of the instance in the stored set that is the closest to the new instance. Gates (1972) comments that a possible decrease in CNN efficiency (due to reduced volume of stored instances) may be compensated by the smaller storage space required by the algorithm, as well as by the shorter classification time required, considering it searches for the 'similar' instances to the one to be classified in a much smaller set of instances.

The Reduced Nearest Neighbor (RNN) algorithm was proposed by Gates (1972) as a refinement of the CNN, which still maintains the same goal i.e., to reduce the set of instances to be stored. The RNN has, as its first step, the execution of the CNN algorithm having as input the original set of instances. In the second step of the RNN, the reduced set returned by the CNN is further processed aiming at a new reduction. The algorithm known as Selective Nearest Neighbor (SNN), proposed by Ritter, Wooddruff, Lowry, & Isenhour (1975), searches for a selective subset of the original set of instances that satisfies the following conditions: (1) it must be consistent (2) all instances of the original set must be nearer to a selective neighbor of the same class than to any instance of the other class and (3) it must be the smallest possible subset. As pointed out by the authors, the condition (2) is the main difference between SNN and CNN, considering that condition (2) for the CNN can be stated as: all instances of the original set must be nearer to a condensed neighbor of the same class than to any condensed neighbor of the other class.

IBL algorithms are in general very sensitive to a few aspects involved in the learning task that may interfere in the learning itself, such as the lack of representativeness of some of the attributes that describe the training instances (superfluous or redundant), the presence of outliers in the training set, the presence of noisy or missing attribute values in the descriptions of instances and, particularly, the distance function chosen for measuring similarity between data instances. As broadly accepted, however, in a ML based environment the use of any learning algorithm is approached combined with the previous use of a data preprocessing process, in charge of dealing with some of the problems related to raw data.

3.2 Collecting and Preprocessing the DataI

In ML applications based on real data, such as educational data, the volume of the training set can be substantial, considering both dimensions of the set i.e., vertical, related to the number of data instances, and horizontal , related to the number of attributes that describe the data instances.

High numbers in either dimension can be the source of problems for ML algorithms. So, when data sets are vertically extensive, usually sampling techniques are used for reducing their number of instances. When they are horizontally extensive, algorithms for the selection of relevant attributes can be employed. Since the PDE system proposal (Section 4) considers an ensemble of IBL algorithms, where some of them have been designed for vertical reduction, sampling has not been considered in the PDE proposal.

Many factors affect the success of ML algorithms. Without doubt the quality of the training instances is the most relevant of all. If the available data are suitable for ML, the task of generalizing the concept can become easier and can be achieved with a lower computational cost involved, when irrelevant or redundant attributes that describe the data instances are removed in a preprocessing phase, prior to automatic learning.

In the context of ML, the problem of selecting subsets of attributes i.e., the selection of attributes that play an important role in characterizing the concept to be learned, has received a lot of attention from the scientific community, particularly since the 1990s. Given a training set that is usually described as a set of instances, each represented as a vector of attribute-value pairs and an associated class (the latter not always present), an attribute selection algorithm, usually employed in a phase prior to training, seeks to identify attributes that are irrelevant or redundant for the description of the concept represented by the training set.

It can be found in the literature several algorithms that aim at reducing the number of attributes (features) that are irrelevant or redundant, in a given set of data instances. Usually such attributes tend to interfere negatively in data processing processes, particularly those with focus on automatic learning. A ML system fed with data described by redundant or irrelevant attributes, most likely will induce expressions involving such attributes, which will probably result in inaccurate generalizations.

Feature selection (FS) algorithms can be helpful for selecting attributes that are relevant for describing the concept embedded in the data set. FS algorithms can be organized in many different ways. One way is by taking into account the evaluation strategy of a selected subset, which measures the effectiveness of the chosen subset.

According to this strategy (John, Kohavi, & Pfleger, 1994), algorithms can be organized in three groups, namely: (1) *filters*: based on the general characteristics of the training set, selecting some features and exclude others; features are filtered independent of the induction algorithm chosen; (2) *wrappers*: involves the use of a ML algorithm to estimate the quality of subsets of attributes and (3) *built-in*: characterized by the feature selection process be an inherent part of the inductive learning algorithm of choice.

Currently, the filter approach is far the most popular due to the increasing number of applications that deal with high volumes of data, for which, the wrapper approach is not feasible; this fact has motivated the use of filter algorithms when defining the PDE's software structure.

Among the several filter algorithms available in the literature, the PDE's approach considers: (1) the Relief algorithm (Kira & Rendell, 1992), with its variants, Relief-A, Relief-B, Relief-C, Relief-D, Relief-E and Relief-F (Kononenko, 1994); (2) the Focus algorithm (Almuallim & Dietterich, 1991) with emphasis on the Focus-2 variants (Almuallim & Dietterich, 1994), C-Focus (Arauzo, Benitez, & Castro, 2003) and C-Focus-3 (Nicoletti & Santoro, 2008). Such algorithms can be of help for removing irrelevant attributes i.e., attributes that do not contribute for characterizing the class associated with instances.

Besides the use of feature selection algorithms, it is also important that the available data input to the PDE system do not have typos, missing attribute values, out-of-range values and contradictory data instances. So, not only the data representation used should be representative of the concept to be learnt, but also the quality of the data instances should be assured i.e., instances described by the same attribute values and belonging to different classes (in this case, dropout and persevere) should be treated separately, associating to them a weight defined by their frequency in the data. The associate weight of such instances can then be used during the classification process as a way to choose one, between the classes. However in many situations the best way to deal with contradictory instances is to remove them from the data set.

4. The Predicting Dropout Events (PDE) Computational System Proposal

The brief survey of models described and the several research work cited in Section 2 can be used as an initial approach to identify relevant aspects of the dropout problem, mainly those related to data i.e., the relevant variables that have impact on the dropout decision, their range of values and their many interrelations, aiming at the design/implementation of the computational system for predicting dropout.

As pointed out by Nicoletti (2019) in relation to computational systems implementing theoretical dropout models available in the literature, "In a very simplified description such systems simulate the dropout/persevere process experienced by the student by implementing an algorithm which, based on the values of a set of input variables, calculates the values of the many other variables that define the other modules of the model to, finally, come to the final decision of dropout or persevere". As reported in the same reference, "Predictive models are strongly dependent on reliable, representative and relevant (to the prediction process) data. […] several dropout models tend to have a loose and general specification of both, the involved variables and the processes that use these variables […].Taking that into account, models are useful as general guidelines for investigating the dropout/persevere process that happens in an educational institution but hardly can be effective for supporting the development of computational systems that implement them.”

As previously stated, the PDE computational system proposal, focus of this article, and described in this section aims at monitoring students’ progress throughout their university academic years, with the intent of detecting, as soon as possible, potential cases of dropout decision, still within a period of time for implementing preventive measures, so to avoid that such decision will be taken. The PDE can be approached as a computational environment having three main subsystems, each in charge of one of the three main tasks: (1) Preprocessing the data of interest and (2) Using IBL algorithms for learning classifiers and (3) Monitoring/Predicting student candidates prone to undergo dropout events. Figure 2 presents a simplified flowchart of the software architecture of the PDE system. For the purpose of collecting relevant and reliable data associated with the several aspects of the academic-social student life that contribute to the persevere/dropout decision, it is mandatory, first:

(a) to define the relevant variables/parameters involved in that decision,

(b) to establish the interval of possible values associated to each identified variable/parameter,

(c) to select the main processes the identified variables/parameters (a) are used,

(d) to determine the possible interrelations between the selected processes.

Such definitions and specifications can be conducted by those involved in the course itself i.e., experienced lecturers and experienced academic staff. This is a difficult task to be conducted, since it is particular to each undergraduate course, as well as dependent of the knowledge area the course encompasses; such definitions and specifications can be highly volatile, considering the constant changes that happen in curricular grids of most courses. Also, for an IBL learning subsystem be feasible and reliable, data instances should be collected over a period of time not shorter than the regular duration of the course.

With focus on a particular higher education course, the PDE system proposal assumes that a set of data instances of interest, representing past records of students that attended such course, is available. Such records are assumed to be described by the same set of attributes; however, when dropouts happen, that may not be the case. Also, each student ́s record should inform the final result obtained by the student in question i.e., failed, succeeded or evaded. The gathering of such set of records is not an easy task to be accomplished, considering it involves many decisions at both levels, administrative and software and, also, considering that some information in these records may be subjected to confidentiality issues. There is also the time factor to be taken into account, remembering that such records over the years may not have been described by the same set of attributes. Decisions must be taken aiming at equalizing them in relation to their descriptions.

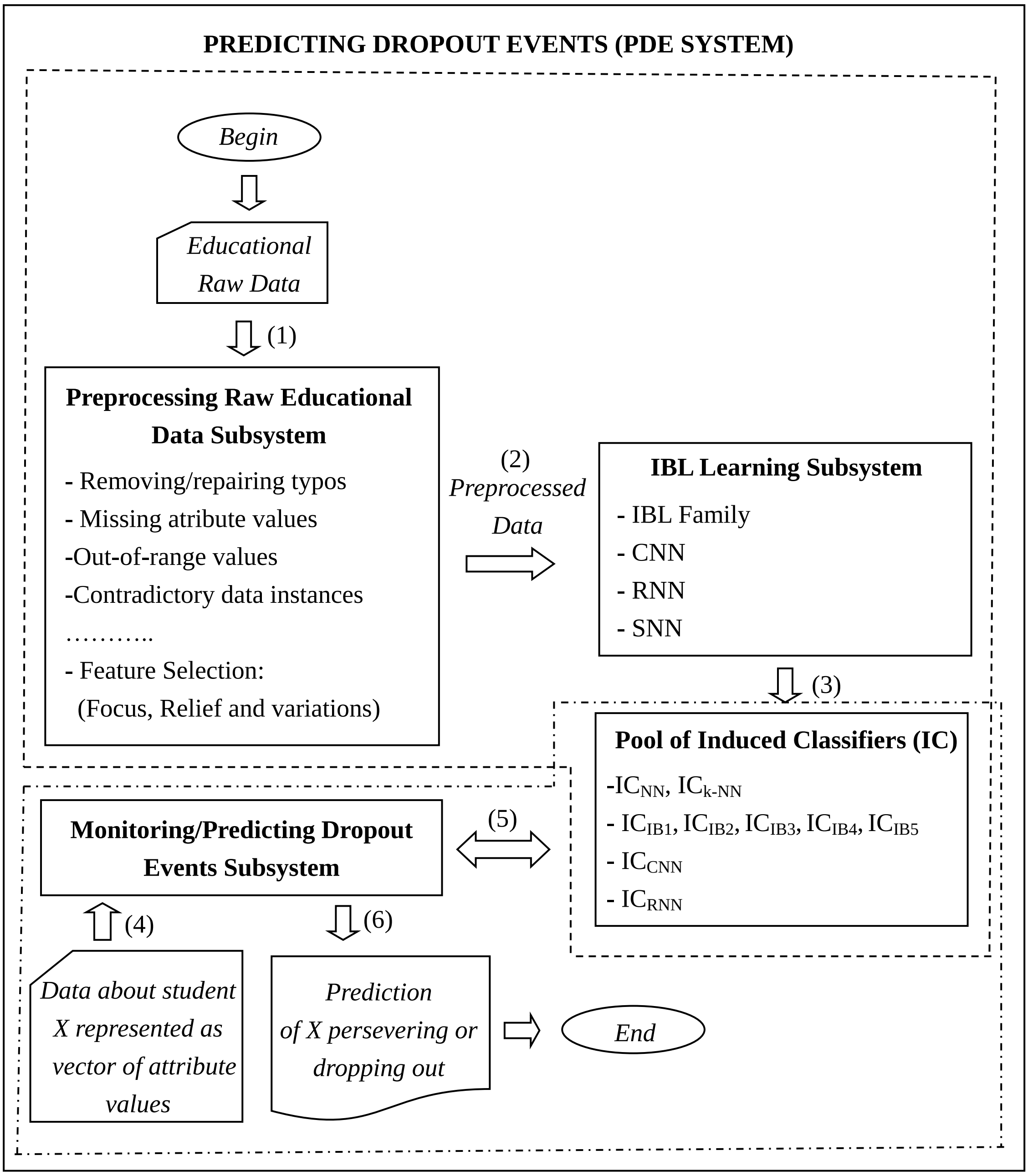
PREDICTING DROPOUT EVENTS (PDE SYSTEM)
Begin
Educational Raw Data
(1)
Preprocessing Raw Educational Data Subsystem
- Removing/repairing typos
- Missing atribute values
-Out-of-range values
-Contradictory data instances
………..
- Feature Selection:
(Focus, Relief and variations)
(2)
Preprocessed Data
IBL Learning Subsystem
- IBL Family
- CNN
- RNN
- SNN
(3)
Pool of Induced Classifiers (IC)
-IC$_{NN}$, IC$_{k\text{-}NN}$
- IC$_{IB1}$, IC$_{IB2}$, IC$_{IB3}$, IC$_{IB4}$, IC$_{IB5}$
- IC$_{CNN}$
- IC$_{RNN}$
Monitoring/Predicting Dropout Events Subsystem
(5)
(4)
(6)
Data about student X represented as vector of attribute values
Prediction of X persevering or dropping out
End

Figure 2. PDE system and its three subsystems: Preprocessing, Learning and Monitoring/Predicting Dropout

The Preprocessing subsystem is in charge of preparing the collected raw educational data to be used by the Monitoring/Predicting subsystem. The Preprocessing subsystem should provide options for the execution of the several tasks described in Section 3.2, including feature selection, related to 'preparing' the data for further use by the other two subsystems. The Learning subsystem implements IBL algorithms for inducing classifiers and the Monitoring/Predicting subsystem uses the induced classifiers for monitoring and detecting possible dropout events. The Monitoring/Predicting subsystem, in a simplistic description, simulates the dropout/persevere process that a student X, who is 'known' (i.e., described) to the subsystem as a vector of attribute values may go through. To do that, the subsystem uses the knowledge induced by the Learning subsystem.

When the Monitoring/Predicting subsystem is in a predicting mode, the vector of attribute values that represents student X is compared with other vectors representing records of students who succeed, failed or dropout in the classifiers induced by the Learning subsystem, aiming at finding the instance most 'similar' to the one that represents student X. If Y is such an instance, X is assumed to adopt Y's decision, be it of success, failure or dropout. Note that instead of using only one IBL classifier from the pool (such as the NN, for example), a committee of various classifiers can be used and the final decision can be reached by the committee by using a frequency-based approach.

5. Discussion and Final Remarks

This paper has its focus on the description of a proposal for a general architecture of a computational system based on the combination of relevant CS-related knowledge mainly from Programming, Software Engineering and Artificial Intelligence (AI), aiming at predicting dropout events in higher-level education. The proposal can be useful in a CS-related teaching environment, for the development of students' skills in the three major areas involved and, specifically, in the two subareas of AI, Machine Learning and Knowledge Representation. The contents of this article can also be of help for promoting students' experience in software development in the context of instance-based learning, by means of investigating slightly different algorithms for implementing instance-based learning and, also, strategies for combining them in an ensemble.

The machine learning based system proposal can also be approached as the general design of a computational system for predicting dropout events not only in higher but also in primary and secondary educational environments (Sansone, 2019; Chung & Sunbok, 2018). As such, when implemented, it can contribute to an earlier detection of possible dropout events and trigger preventive strategies for avoiding that to happen. The proposal can also be helpful for the whole educational environment, independent of the course and could be adapted for other types of educational environments, such as e-learning.

As stressed throughout the paper, it cannot be forgotten that the outcome of a predictive computational system able to detect dropout/persevere events will be reliable if the system reflects real situations and, also, it is based on a group of variables and parameters that effectively contribute to characterize and predict an event of dropout, in time to be dealt with before it happens.

The fact of using IBL algorithms for predicting dropout events, instead of algorithms that generalize the expression of the concept in the training phase, can be quite a convenient choice for this type of computational application, since the process such algorithms implement are easily understood by the vast majority of their main users (i.e., administrative staff) and it is easily implementable. As pointed out by Nicoletti (2019), "It is also mandatory to have all the available information about each relevant variable i.e., how it is used, the nature of its associated value (permanent or transient), its range of values and its degree of relevance related to each process that uses it. It should also be considered that many of the variables involved in the dropout/persevere decision process are very volatile in the sense that their current values may suddenly change as the result of some unpredicted instability and, as consequence, their corresponding degree of relevance in the process may change as well."

Acknowledgments

The first author is grateful to CNPq for the research grant PQ 2019. Both authors are grateful to UNIFACCAMP. This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brazil (CAPES) – Finance Code 001.

References

Aha, D. W., Kibler, D., & Albert, M. K. (1991). Instance-based learning algorithms. *Machine Learning, 6*, 37-66.

Aha, D. W. (Ed.) (2013). Lazy Learning. Springer Science+Business Media Dordrecht.

Aljohani, O. (2016). A comprehensive review of the major studies and theoretical models of student retention in higher education. *Higher Education Studies, 6*(2), 18. https://doi.org/10.5539/hes.v6n2p1

Almuallim, H., & Dietterich, T. G. (1991). Learning with many irrelevant features. In Proc. of the 9th National Conference on Artificial Intelligence, pp. 547-552.

Almuallim, H., & Dietterich, T. G. (1994). Learning Boolean concepts in the presence of many irrelevant features. *Artificial Intelligence, 69*(1-2), 279-305. https://doi.org/10.1016/0004-3702(94)90084-1

Arauzo, A., Benitez, J. L., & Castro, J. L. (2003). C-Focus: a continuous extension of Focus. In J. Benitez, O. Cordón, F. Hoffmann & R. Roy (Eds.), *Advances of Soft Computing – Engineering Design and Manufacturing* (pp. 225-232). Springer-Verlag. https://doi.org/10.1007/978-1-4471-3744-3_22

Bean, J. P. (1980). Dropouts and turnover, The synthesis and test of a causal model of student attrition. *Research in Higher Education*, *12*(2), 155-187. https://doi.org/10.1007/BF00976194

Bean, J. P. (1985). Interaction effects based on class level in an explanatory model of college student dropout syndrome. *American Educational Research Journal, 22*(1), 35-64. https://doi.org/10.3102/00028312022001035

Bean, J. P., & Metzner, B. S. (1985). A conceptual model of nontraditional undergraduate student attrition. *Review of Educational Research, 55*(4), 485-540. https://doi.org/10.3102/00346543055004485

Belloc, F., Maruotti, A., & Petrella, L. (2010). University drop-out: an Italian experience. *High Education, 60*(2), 127-138. Retrieved from https://link.springer.com/article/10.1007/s10734-009-9290-1

Chrysikos, A., Ahmed, E., & Ward, R. (2017). Analysis of Tinto ́s student integration theory in first-year undergraduate computing students of a UK higher education institution. *International Journal of Comparative Education and Development, 19*(23), 97-121. https://doi.org/10.1108/IJCED-10-2016-0019

Chung, J. Y., & Sunbok, L. (2018). Dropout early warning systems for high school students using machine learning. *Children and Youth Services Review, 96*, 346-353. https://doi.org/10.1016/j.childyouth.2018.11.030

Cover, T. M., & Hart, P. E. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory, 13*, 21-27. https://doi.org/10.1109/TIT.1967.1053964

Dekker, G. W., Pechenizkiy, M., & Vleeshouwers, J. M. (2009). Predicting students drop out: a case study. *Educational Data Mining*, 41-50. Retrieved from http://www.educationaldatamining.org/EDM2009/uploads/proceedings/dekker.pdf

Demetriou, C., & Schmitz-Seiborski, A. (2009). Integration, motivation, strengths and optimism: retention theories past, present and future. In Proc of the 7th National Symposium on Student Retention, pp. 300-312. Retrieved from https://studentsuccess.unc.edu/files/2012/11/Demetriou-and-Schmitz-Sciborski.pdf

Durso, S. O., & Cunha, J. V. (2018). Determinant factors for undergraduate student ́s dropout in an accounting studies department of a Brazilian public university. *Educa ção em Revista (EDUR), 34*, e186332. http://dx.doi.org/10.1590/0102-4698186332

Ethington, C. A. (1990). A psychological model of student persistence. *Research in Higher Education, 31*(3), 279-293. https://doi.org/10.1007/BF00992313

Gates, G. H. (1972). The reduced nearest neighbor rule. *IEEEE Transactions on Informatin Theory, 18*(3), 431-433. https://doi.org/10.1109/TIT.1972.1054809

Giannakos, M. N., Aalberg, T., Divitini, M., Jaccheri, L., Mikalef, P., Pappas, I. O., & Sindre, G. (2017). *Identifying dropout factors in information technology education: a case study*. In Proc, of 2017 IEEE Global Engineering Education Conference (EDUCON), pp. 1191-1198. http://dx.doi.org/10.1109/EDUCON.2017.7942999

Gordon, N. A. (2016). *Issues in retention and attainment in Computer Science*. Higher Education Academy, University of Hull, pp. 23. Retrieved from https://www.heacademy.ac.uk/knowledge-hub/issues-retention-and-attainment-computer-science

Hart, P. E. (1968). The condensed nearest neighbor rule. *IEEE Transactions on Information Theory, 14*, 515-516. https://doi.org/10.1109/TIT.1968.1054155

John, G., Kohavi, R., & Pfleger, K. (1994). *Irrelevant features and the subset selection problem*. In Proc. of the International Conference on Machine Learning, pp. 121-129. https://doi.org/10.1016/B978-1-55860-335-6.50023-4

Kerby, M. (2015). Toward a new predictive model of student retention in higher education: an application of classical sociologval theory. *J. College Student Retention Research Theory and Practice, 17*(1). https://doi.org/10.1177/1521025115578229

Kira, K., & Rendell, L. (1992). *A practical approach to feature selection*. In Proc. of the 9th International Conference on Machine Learning (ICML), pp. 249-256. https://doi.org/10.1016/B978-1-55860-247-2.50037-1

Kononenko, I. (1994). *Estimating attributes: analysis and extension of Relief*. In Proc. of the European Conference on Machine Learning (ECMA), pp. 171-182. https://doi.org/10.1007/3-540-57868-4_57

Marshland, S. (2009). *Machine Learning An Algorithm Perspective*. USA:Chapman & Hall/CRC Press.

Murray, M. (2014). Factors affecting graduation and student dropout rates at the University of KwaZulu-Natal. *South African Journal of Science, 11*(11/12), 6. https://doi.org/10.1590/sajs.2014/20140008

Nicoletti, M. C. (2019). Revisiting the Tinto's theoretical dropout model. *Higher Education Studies, 9*(3), 52-64. https://doi.org/10.5539/hes.v9n3p52

Nicoletti, M. C., Reali, A. M. M. R., Dias, T. C. M., & Abib, S. (2012). Survey, categorization and analysis of the main causes for dropout from the UFSCar-UAB-Pedagogy Course (per íod: 2007-2011) (in Portuguese). Proc. of the IX Congresso Brasileiro de Ensino Superior a Dist ância (ESUD 2012), pp. 1-14.

Nicoletti, M. C., & Santoro, D. M. (2008). The influence of search mechamisms in feature subset selection processes. *Intelligent Decision Technologies*, *2*(4), 231-1238. https://doi.org/10.3233/IDT-2008-2404

Pascarella, E. T. (1980). Student-faculty informal contact and college outcomes. *Review of Educational Research, 50*(4), 545-595. https://doi.org/10.3102/00346543050004545

Pascarella, E. T., & Chapman, D. W. (1980). Validation of a theoretical model of college withdrawal: Interaction effects in a multi-institutional sample. *Research in High Education, 19*(1), 25-48. https://doi.org/10.1007/BF00977337

Paura, L., & Arhipova, I. (2014). Cause analysis of students' dropout rate in higher education study program. *Procedia – Social and Behavioral Sciences*, *109*, 1282-1286. https://doi.org/10.1016/j.sbspro.2013.12.625

Pidgeon, A. M., Rowe, N. F., Stapleton, P., Magyar, H., & Lo, B. C. Y. (2014). Examining characteristics of resilience among university students: an international study. *Open Journal of Social Sciences, 2*, 14-22. https://doi.org/10.4236/jss.2014.211003

Ritter, G. L., Wooddruff, H. B., Lowry, S. R., & Isenhour. T. L. (1975). An algorithm for a selective nearest neighbor decision rule. *IEEE Transactions on Information Theory, 21*(6), 665-669. https://doi.org/10.1109/TIT.1975.1055464

Rovai, A. P. (2003). In search of higher persistence rates in distance education online programs. *Internet and Higher Education, 6*, 1-16. https://doi.org/10.1016/S1096-7516(02)00158-6

Sansone, D. (2019). Beyond early warning indicators: High school dropout and machine learning. *Oxford Bulletin of Economics and Statistics*, *81*(2), 0305-9049. https://doi.org/10.1111/obes.12277

Smith, J. P., & Naylor, R. A. (2001). Dropping out of university: A statistical analysis of the probability of withdrawal for UK university students. *Journal of the Royal Statistical Society-Series A, 164*, 389-405. https://doi.org/10.1111/1467-985X.00209

Spady, W. (1970). Dropouts from higher education: an interdisciplinary review and synthesis. *Interchange, 1*(1), 64-85. Retrieved from https://link.springer.com/article/10.1007/BF02214313

Spady, W. (1971). Dropouts from higher education: towards an empirical model. *Interchange*, *2*(3), 632-644. https://doi.org/10.1007/BF02282469

Tinto, V., & Cullen, J. (1973). *Dropout in higher education: a review and theoretical synthesis of recent research*. Office of Education (DHEW), Contract OEC-0-73-1409, pp. 99. Retrieved from https://files.eric.ed.gov/fulltext/ED078802.pdf

Tinto, V. (1975). Dropout from higher education: a theoretical synthesis of recent research. *Review of Educational Research, 45*(1), 89-125. https://doi.org/10.3102/00346543045001089

Tinto, V. (1997). Classrooms as communities: exploring the educational character of student persistence. *The Journal of Higher Education*, *68*(6), 599-623. https://doi.org/10.1080/00221546.1997.11779003

Vergidis, D., & Panagiotakopoulos, C. (2002). Student dropout at the Hellenic open university - evaluation of the graduate program: studies in education. *International Review of Research in Open and Distance Learning, 3*(2), 1-15. https://doi.org/10.19173/irrodl.v3i2.101

Willging, P. A., & Johnson, S. D. (2004). Factors that influence students' decision to drop out of online courses. *Journal of Asynchronous Learning Networks, 13*(3), 115-127. https://doi.org/10.4236/jss.2014.211003

Wiseman, R. L., Gonzales, S. M., & Salyer, K. (2004) A cross-cultural analysis of students' sense of community, degree of involvement, and educational benefits. *Intercultural Communication Studies, XIII-1*, 173-190. Retrieved from https://web.uri.edu/iaics/files/14-Richard-L.-Wiseman-Star-M.-Gonzales-Kimberly-Salyer.pdf

Xenos, M., Pierrakeas, C., & Pintelas, P. (2002). A survey on student dropout rates and dropout causes concerning the students in the course of informatics of the Hellenic Open University. *Computers & Education, 39*(4), 361-377. https://doi.org/10.1016/S0360-1315(02)00072-6

Copyrights