Received 14 September 2023, accepted 4 October 2023, date of publication 10 October 2023, date of current version 18 October 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3323202

RESEARCH ARTICLE

A Time-Aware Approach for MOOC Dropout Prediction Based on Rule Induction and Sequential Three-Way Decisions

CARLO BLUNDO, VINCENZO LOIA, (Senior Member, IEEE), AND FRANCESCO ORCIUOLI, (Member, IEEE)

Dipartimento di Scienze Aziendali - Management and Innovation Systems, Università degli Studi di Salerno, Fisciano, 84084 Salerno, Italy

Corresponding author: Francesco Orciuoli (forciuoli@unisa.it)

ABSTRACT Nowadays, Massive Open Online Courses (MOOCs) are adopted by students worldwide. One of the main critical issues often associated with MOOCs is the dropout phenomenon. In other words, the percentage of students abandoning a MOOC-based study path is considered still too high. Therefore, an increasing number of scientific works, coming from several heterogeneous communities (e.g., computer science, data science, statistics, education) propose approaches trying to mitigate such a problem. The majority of the aforementioned works focus on machine learning methods to define classifiers able to be trained and, subsequently, to predict students who are going to abandon a course before it ends. Among such approaches, the ones achieving the best performance use enriched sets of features (to train their models) and produce results that cannot be used to easily clearly characterize the different behaviors of dropping-out and non-dropping-out students. The present work proposes the design of a novel process to train a set of dropout predictors leveraging on a reduced set of features. The underlying idea is to exploit weekly data in order to classify, with acceptable levels of precision, students who are likely going towards dropout or not. In cases of uncertainty, the classification decision is deferred to the next week, when new data is available. Such an approach, which takes care and is aware of the course timeline, offers several advantages. The first one is the chance to build a real-time educational decision support system able to support decision as sufficient information is available (as the time goes on). The second one is to preserve resources and avoiding wasting them with students erroneously classified at risk of dropout. The third one is to allow explicit characterization of dropout-conducing behavior by using a rule mining approach.

INDEX TERMS Dropout predictions, sequential three-way decisions, rough set theory, rule induction.

I. INTRODUCTION

In recent years, many students worldwide are progressively considering Massive Open Online Courses (MOOCs) as learning platforms, which help them to fill the gaps or acquire new skills required by the industry [1]. During the COVID-19 pandemic, the interest in such courses has seen exponential growth due to people in a great part of countries worldwide going into long quarantine lockdown [2]. However, a great percentage of students who enroll in a MOOC fail to complete the course [3]. Indeed, one of the major problems affecting MOOCs is the great number of students abandoning courses [4], a phenomenon extensively studied by researchers in different fields by proposing works that analyze such phenomenon and also describe approaches trying to mitigate it.

The scientific literature agrees that early prediction of students tending to dropout is vital to allow staff intervention at the right time, providing them with extra help, such as additional course resources, peer support, and teacher guidance, improving also the effectiveness and quality of online learning [5].

The associate editor coordinating the review of this manuscript and approving it for publication was Kostas Kolomvatsos.

The dropout phenomenon has been addressed in several study areas where different solutions have been proposed to mitigate the problem. Some of these studies, instead of using traditional models to make predictions, work on the characterization of the student's behavior [6] defining a well-structured feature engineering process starting from data stored in MOOC platform log files [7]. In [4] the authors state that the current predictive models take into consideration simple events such as videos, and exercises, but these models may have better performance if they include more complex learners' behaviors. Indeed, the authors propose several independent variables based on events, demographic data, intentions, videos, and exercises. Furthermore, the authors divide the course data log into five weeks to find the best moment to predict dropout students and their results show that is possible to obtain a good AUC level (greater than 0.9) from the second week using a Random Forest model.

The main contribution of the present work is to analyze the dropout-leading student's behavior along the time dimension where the temporal line is divided into a sequence of slots of fixed length. At the end of each slot, a classifier is trained to support dropout predictions at that stage. The classifiers are trained through the application of the CN2 Rule Induction algorithm [8], thus the training result is a set of human-readable rules. In other words, the behavior of dropout students can be easily characterized by using the expressive representation of such rules. The most important aspect of the proposed training process is its capability to train adequate-performance classifiers by adopting a Sequential Three-way Decisions approach [9], i.e., it is able to defer classification decisions to the next iteration (considering an additional slot of data) for those students whose behavior cannot be classified as dropout leading or non-dropout leading within an acceptable degree. During the next iteration, deferred decisions could be taken cause more knowledge is available (an additional slot of data). During the training process, decision deferment is realized by excluding a given sample from the training set when such a sample is included in the boundary region of the tri-partitioning, i.e., when the information of such a sample is not sufficient to consider the sample as a dropout or non-dropout.

The proposed approach unfolds along three main steps. The first two steps borrow and enhance the results provided in [10] while the last step is completely new and original. The aforementioned steps are: i) Employing a reduced set of features comprising only event-based variables (derived by processing data obtained from common log files) allowing its applicability to any MOOC platform. ii) Engineering the aforementioned features by aggregating log file data into a sequence of time slots (e.g., week by week) obtaining a time series of multidimensional features for improving the results of the classification task. iii) Designing a training process for building a set of classifiers to support an iterative/incremental classification task. At the end of each time slot, these classifiers can either make a decision (dropout or non-dropout) if sufficient information is available or defer the decision to the next iteration (at the end of the subsequent time slot) if information is insufficient. The proposed approach offers results whose performances are higher than those of existing methods exploiting the same or a similar set of features.

The main advantage of the approach is the chance to anticipate decisions (predictions about dropout) for those students for whom it is possible to collect sufficient information during the first periods of the course and defer to the next periods such decisions in the case of uncertainty (insufficient information). The capability to be aware of course time concretely sustains the definition of a real-world educational decision support system.

Experimentation activities were conducted over a public dataset from the Chinese MOOC platform XuetangX (adopted in the context of KDD Cup 2015[1]) and showed interesting results, both quantitative and qualitative. Lastly, the approach could be scaled in context where users' behaviors have to be determined by using activity-based log files.

The remaining part of the paper is organized as it follows. Section II provides a review of literature dealing with the dropout phenomenon. Section III describes the main steps related to data preparation starting from the characteristics of the considered data sources. Section IV presents the overview of the approach and the main components of the system architecture. Section V deals with a detailed description of the training process, providing also background knowledge related to Sequential Three-Way Decisions and CN2 Rule Induction. Section VI describes how the built classifiers can be injected into an educational decision support system to provide dropout predictions. Section VII describes the experimentation and evaluation activities conducted with a well-known dataset. Lastly, Section VIII provides considerations about the obtained results and anticipates possible future works.

II. RELATED WORKS

The focus of this section is on the scientific literature related to at-risk students' dropout prediction on MOOC platforms. In particular, the authors of [4] defines seven categories of variables that can be used to train dropout predictors: self-reported strategies, demographic variables, variables describing intentions, event-based features, activities, videos and exercises. The first three categories could be obtained from a questionnaire and the other four should be traced during students' interactions with MOOC platforms. The defined variables are all specific and often fine-grained. Typically, only event-based features can be extracted from the log files of the vast majority of MOOC platforms (time spent on a page, number of visited pages, etc.) but authors of [4] used more specific variables (e.g., related to interaction patterns with videos and assessment which are not always available). Activities, videos and exercises

[1] www.kddcup2015.com.

include detailed information about the students' interactions with the educational resources. It is possible to indicate such categories as activity-based features. Such authors assert that variables related to specific event-based features when combined to exercises, videos or activities variables can provide the best results (AUC of 97%, 96%, and 96%) for dropout predictions through the Random Forest model. Moreover, they obtained AUC equal 52% by applying the same model only to self-reported strategies. In further work, the authors of [11] state that motivation and engagement are critical factors influencing the students' dropout-conducing behavior. In particular, they obtain a set of features, from the event-based variables that they link to the levels of engagement and motivation of students but they do not propose a prediction model. Some examples of the aforementioned variables are the total time of the active sessions, the average session time, etc. Even other works, like [12], train classifiers by using event-based features like, for instance, the number of contributions to course discussion threads, total time of activity, total time of inactivity, etc. The same authors define a window of fifteen minutes from the start of the user's log record and any activity registered within the time window is considered part of one session. The authors obtain the best result of AUC 89, 56% with a Gradient Boost Decision Tree model.

Other works, recognized in the specialized literature, adopt a subset of the aforementioned features to predict students' dropout-conducing behavior. In [13] the authors propose a temporal prediction-based deep learning model not only focused on predicting learners' dropout but also used to plan the intervention to mitigate such risk. The model is based on traditional and quantitative variables in the category of event-based features such as the number of access to the platform, the number of quizzes, the number of active days, and so on. This model obtains AUC equal to 96.5%. Other works dealing with time for dropout prediction are [14] and [15]. The latter proposes a Time-Missing-Aware LSTM unit to mitigate the impact of data sparsity by integrating informative missingness patterns into the model. Moreover, the authors of [5], whose approach to feature engineering also considers time dimension, achieve an accuracy of 87% for the classification task.

III. FEATURE ENGINEERING

As already stated, the proposed approach foresees only the use of event-based features that can be derived from the processing of information included in the log files of any MOOC platform. The feature construction process has already been defined in the previous work [10] and, in order to explain it, the structure of a typical log file for a Web application needs to be introduced.

Fig. 1 depicts a sample construction of a set of timed features starting from a log file. In particular, the log file provides records representing the information related to the interaction of a student with the resources of an online course. Each record of the log file is structured as it follows: `enrollment_id` (indicating a given student interacting with the resources of a given course), `time` (indicating the time at which the interaction starts) and `event` (indicating the specific event related to the interaction like, for instance, accesses to course pages, problems, videos, wiki pages, discussion forums, etc.). The feature construction process is realized by executing three steps. The first step is to divide the whole course period into an ordered sequence of n timeslots. The second step is to select the k events of interest. The third step is to aggregate into a unique `enrollment_id` value and for each timeslot such k events. In this way, for each pair (student, course), it will be created k aggregated values for each timeslot. The result of a sample application of the feature construction process is reported in Fig. 1. In such an example, the length of each timeslot is 1 day and a simple counting operation has been used as an aggregation function. Moreover, it is possible to note that, for the `enrollment_id` equal to 3453, three videos have been played during the first timeslot, one video during the second timeslot, and so on. The result of the feature construction process provides a set of features describing the behavior of a given student, interacting with a specific online course, over time. In particular, the set of the constructed features is:

$$A = A^1 \cup A^2 \cup \cdots \cup A^n \tag{1}$$

where A^j is the set of features in timeslot $j = 1, \ldots, n$. Moreover, the set A^j could be detailed as it follows:

$$d_1^j \cup d_2^j \cup, \ldots, \cup d_k^j \tag{2}$$

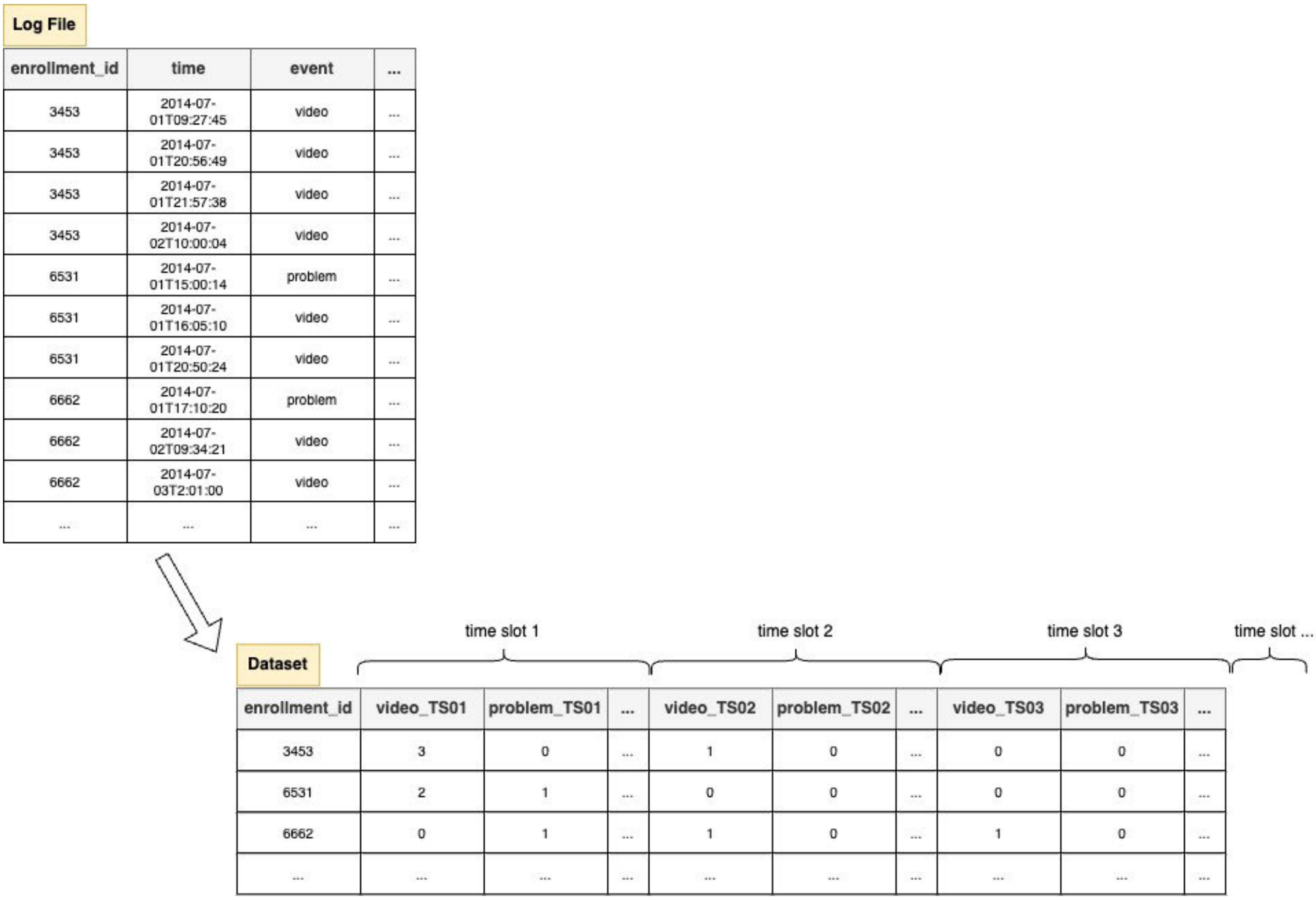
where d_i^j is the i-th feature in timeslot j, with $i = 1, \ldots, k$.

IV. OVERALL APPROACH

The main idea underlying the proposed approach is that the decision process, classifying students (interacting with specific courses) to exactly one of the dropout and non-dropout classes, can be deployed as an incremental/iterative process in a way that in case of lack of information, the classification decision can be deferred to the next iteration.

Fig. 2 graphically describes the main idea of the proposed decision process. Once a timeslot is completed, the log file records included in such timeslot are processed to obtain the feature values for such timeslot. Therefore, the set of such values is added to the ones associated to the previous timeslots. Using such values, the predictor will decide if a specific student is going to abandon a given course, if she/he will retain or if the decision cannot be taken cause lack of knowledge. In the last case, the student for whom the decision has been deferred will be considered during the next decision process iteration when feature values for the outcoming timeslot have been processed (this could mean more knowledge to consider to make the decision).

In order to realize the aforementioned idea, two processes have been defined. The first one is the *training process* where a training dataset is exploited to build a set of classifiers that are used in the *prediction process* to effectively predict dropout or non-dropout for the new incoming data.



Log File

enrollment_id	time	event	...
3453	2014-07-01T09:27:45	video	...
3453	2014-07-01T20:56:49	video	...
3453	2014-07-01T21:57:38	video	...
3453	2014-07-02T10:00:04	video	...
6531	2014-07-01T15:00:14	problem	...
6531	2014-07-01T16:05:10	video	...
6531	2014-07-01T20:50:24	video	...
6662	2014-07-01T17:10:20	problem	...
6662	2014-07-02T09:34:21	video	...
6662	2014-07-03T2:01:00	video	...
...	...	...	...

Dataset

	time slot 1			time slot 2			time slot 3			time slot ...
enrollment_id	video_TS01	problem_TS01	...	video_TS02	problem_TS02	...	video_TS03	problem_TS03	...	
3453	3	0	...	1	0	...	0	0	...	
6531	2	1	...	0	0	...	0	0	...	
6662	0	1	...	1	0	...	1	0	...	
...	...	...	...	...	...	...	...	...	...	

FIGURE 1. Feature construction.

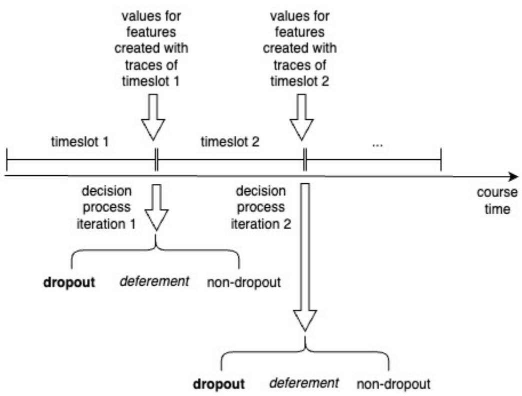


FIGURE 2. Iterative/incremental decision process.

More in detail, Fig. 3 shows the workflow of the training process. As shown by the figure, a labeled training set (features A_i^j are generated as indicated in Section III and feature d contains value 1 in case of dropout or 0 in case of non-dropout) is iteratively/incrementally processed by using a modified version of Sequential Three-Way Decisions [16] algorithm, based on Probabilistic Rough Sets, that is used to execute a tri-partitioning including three regions: POS, BND, and NEG. Each row of the training dataset represents an object containing data related to a given student interacting with a specific course. Each object is described along the timeline that is divided into timeslots as explained before. Therefore, region POS contains all students who are surely going to abandon a given course. Region NEG contains all students who certainly will not abandon a given course. Lastly, BND is a gray region in which it is possible to find students for whom there is uncertainty. Once the tri-partitioning has been obtained, a classifier is built by using CN2 Rule Induction algorithm. In particular, at the j-th iteration the built classifier predicts three target classes (POS, BND and NEG) for the objects. Moreover, the objects in the BND region are further processed during the $j+1$-th iteration after that their descriptions are enriched with features A^{j+1}. In this way, during the $j+1$-th iteration, a new classifier is built for predicting dropouts for the uncertain cases found in the j-th iteration. The main advantage of this approach for building classifiers is that the decisions related to uncertain cases are deferred until new information arrives to fill the lack of knowledge.

Once the set of n classifiers is ready (n is the number of available timeslots), such classifiers can be used during the prediction process to make real-time predictions related

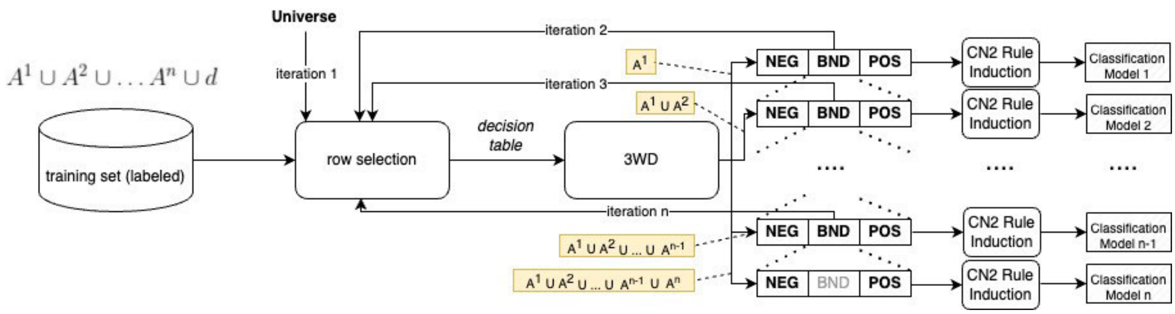


FIGURE 3. Training process.

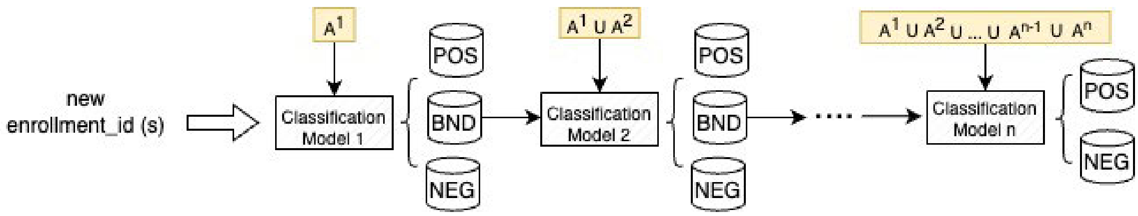


FIGURE 4. Prediction process.

to new data. Fig. 4 shows how the ordered sequence of classifiers has been applied using a number larger and larger of features. Such number increases as a new timeslot ends and new traces are processed and used to derive features. It is important to note that, at the end of each iteration, objects classified as BND are used as input for the next classifier. The last classifier of the sequence is trained with two target classes given that a decision must be necessarily taken. In other words, assume to be at the first iteration (and of timeslot 1), at this stage the description of the object x contains features A^1. On such features, the rules of classification model 1 have been applied. Thus x is classified in exactly one of the three regions. Assume that x is classified as BND. In this case, x is shifted to the second stage (at the end of timeslot 2), its description is enriched with features derived from traces of timeslot 2 and the considered set of features is now $A^1 \cup A^2$. Classification Model 2 is applied and if the predicted class is POS or NEG, x is no longer active along the iterations of the prediction process but a specific strategy can be executed by the staff to mitigate the risk of dropout in case of POS or no action is undertaken in case of NEG.

V. TRAINING METHOD

The training method is defined by integrating Probabilistic Rough Sets and CN2 Rule Induction into the Sequential Three-Way Decisions.

A. PROBABILISTIC ROUGH SETS

Rough Sets (RS) have been introduced by Zdzislaw Pawlak [17] as an extension of set theory for the study of intelligent systems characterized by imprecise information. A rough set is a formal approximation of a conventional crisp set in terms of a pair of sets that give the lower and the upper approximations of the original set. A key concept of RS is the indiscernibility relation. This is a binary relation that expresses the fact two objects are indiscernible (or indistinguishable) on the basis of their descriptions. Let us define formally this concept. A limitation of traditional RS theory is that it does not allow any uncertainty in the definition of lower and upper approximations and, to introduce a certain degree of tolerance in the definition of the two approximations, the probability approximation space was brought into RS [18].

More in detail, an Information System is a pair $IS = (U, A)$ where the non-empty finite set U is the universe of discourse and A is a non-empty finite set of attributes describing the objects $x \in U$, and A is such that: $a : U \rightarrow V_a$, where V_a is a set of values of attribute a, called domain of a. IS can be represented by a data table where rows are the objects and columns are the attributes. If $A = C \cup D$ where C is a non-empty set of condition attributes and D is a non-empty set of decision attributes, an Information System is called Decision System,

An indiscernibility relation I_B on U, where $B \subseteq A$, is defined as it follows:

$$I_B = \{(x, y) : x, y \in U \wedge \forall b \in B, b(x) = b(y)\}. \quad (3)$$

Once defined I_B it is possible to create a partition of U where each part is an equivalence class based on I_B:

$$[x]_B = \{y \in U : (x, y) \in I_B\}. \quad (4)$$

All the equivalence classes (called also information granules) $[x]_B$ form the blocks of the partition U/B. Changes to B lead to changes in the composition of granules.

Therefore, let $X \subseteq U$ a subset of the Universe, $B \subseteq A$ a subset of attributes and I_B an indiscernibility relation I_B on U, it is possible to define the lower and upper approximation operators based on Probabilistic Rough Sets (PRS) by using the conditional probability, $P(X|[x]) = \frac{|X \cap [x]|}{|[x]|}$:

$$\underline{B}(X) = \{x \in U : P(X|[x]_B) \geq \alpha\}, \tag{5}$$
$$\overline{B}(X) = \{x \in U : P(X|[x]_B) > \beta\}, \tag{6}$$

where α and β ($0 \leq \beta < \alpha \leq 1$) are two thresholds and establish the tolerance degree used to determine both lower and upper approximations.

B. THREE-WAY DECISIONS

Three-Way Decisions (3WD) theory [19] models a particular class of human ways of problem solving and information processing. The basic idea of such theory is to divide a universal set (of objects) into three pair-wise disjoint regions, or more generally a whole into three distinctive parts, to handle complexity, and to act upon each region or part by developing an appropriate strategy. Models for 3WD can be based on Rough Sets. In particular, Yao proposes, in [9], the application of Probabilistic Rough Set Theory (with a pair of thresholds). More in detail, starting from lower and upper approximations, the three regions can be calculated as it follows:

$$\begin{aligned} POS(X) &= \underline{B}(X), \\ BND(X) &= \overline{B}(X) - \underline{B}(X), \\ NEG(X) &= U - \overline{B}(X). \end{aligned} \tag{7}$$

In the previous equations, $X \subseteq U$ is a concept defined as a subset of the universe of discourse U and (U, A) (remembering that $B \subseteq A$) is an information table/system where A is a set of attributes describing the objects $x \in U$. More in detail, three regions $POS(X)$, $BND(X)$, and $NEG(X)$ respectively represent the sets of i) objects that certainly belong to X, ii) objects for which it is not possible to affirm that they belong to X nor that they do not belong to X, and iii) objects that certainly do not belong to X.

C. CN2 RULE INDUCTION

Rule induction is a data mining process for deducing if-then rules from a data set. These symbolic decision rules explain an inherent relationship between the attributes and class labels in the dataset. CN2 [20] is an algorithm designed for the efficient induction of simple, comprehensible rules adopted for executing the knowledge discovery task. CN2 has been also improved by several works like [8]. In the context of this work, CN2 Rule Induction algorithm is proposed to build time-dependent rule-based systems for the prediction of dropouts through three-way classifiers sequentially trained with an increasing number of features. CN2 algorithm can be applied to discrete values and produces human-readable and understandable rules that could be considered to simply characterize the dropout-conducting behavior of students or, the safe (non-dropout) behavior and the uncertain (borderline) behavior.

D. S3WD-BASED TRAINING ALGORITHM

The training algorithm S3WDT (Sequential Three-Way Decisions Training) described in Section IV is detailed in Algorithm 1. Each iteration of the for loop represents the computation at the end of the corresponding timeslot where a new three-way classifier ($CL[j]$) is trained by means of the `CN2` function and by exploiting the data pre-processing realized by means of the application of the probabilistic three-way decisions function (`3WD`). Such a function is applied on objects still unclassified by using new information obtained during the last timeslot of interactions (with the MOOC platform). Such information growth is the one introduced in Sections III and IV. As it is possible to note, each iteration considers less (or equal) number of objects to process. The `3WD` function is implemented on the basis of Equations (5), (6), and (7). The function `CN2` is the implementation of the CN2 Rule Induction algorithm described in Section V-C.

Algorithm 1 The S3WD-Based Training Algorithm (S3WDT)

```
Input: U, the training data
Input: A = A^1 ∪ ··· ∪ A^n, the set of attributes
Input: β and α, the probabilistic three-way decisions
       thresholds
Output: CL, the list of three-way classifiers
begin
  C ← dropout
  U^0 ← U
  B ← ∅
  for j = 1 to n do
    B ← B ∪ A^j
    (POS^j, BND^j, NEG^j) ← 3WD(U^{j−1}, B, β_j, α_j)
    if j < n then
      CL[j] ← CN2(POS^j, BND^j, NEG^j)
    end
    else
      CL[j] ← CN2(POS^j, NEG^j)
    end
    U^j ← BND^j
  end
end
```

VI. PREDICTION METHOD

The prediction method deploys the classifiers built during the training process and applies them sequentially in order to classify new objects in real time.

A. S3WD-BASED PREDICTION ALGORITHM

The prediction algorithm S3WDP (Sequential Three-Way Decisions Prediction) described in Section IV is reported in

Algorithm 2. Such an algorithm exploits the list of classifiers (CL) built during the training phase.

Algorithm 2 The S3WD-Based Prediction Algorithm (S3WDP)

```
Input: y, the object to be classified as dropout or
       non-dropout
Input: CL, the list of three-way classifiers
begin
    B ← ∅
    for j = 1 to n do
        P ← CL[j]
        B ← B ∪ A^j
        D ← Des(x, B)
        p ← P(y, D)
        if p == POS then
            class ← dropout
            return
        end
        else if p == NEG then
            class ← non-dropout
            return
        end
    end
end
```

The prediction algorithm considers at each iteration (of the for loop) the classifier corresponding to the end of the last timeslot and the data available, at that time, for the object to be classified. Once the classifier produces the predicted class, if it is POS (dropout is predicted) or NEG (non-dropout is predicted) then the loop finishes and the current object is considered classified. If the prediction is BND then uncertainty has been recognized and the classification process continues to the next iteration where information gathered during a new timeslot has been acquired. Such new iteration considers also the next predictor in `P` which is a function receiving a new object and its description and returns the predicted class (`dropout` or `non-dropout`). The function `Des` is the description of the object to be classified along the attribute set B.

VII. EXPERIMENTATION AND EVALUATION RESULTS

Experimentation and evaluation activities were conducted over a public dataset from the Chinese MOOC platform XuetangX (adopted in the context of KDD Cup 2015).[2] The original dataset consists of different *CSV* files [21]. In particular, these files were merged to obtain a single table reporting all the interactions of students with the Web resources of the courses they are enrolled in. The table has been enriched with truth labels (1 for dropout and 0 for non-dropout associated to all rows) and subsequently processed as indicated in Section III where the following 13 features have been generated for each one of the 6 considered timeslots: *number of problems accessed*, *number of videos played*, *number of accesses to the course material*, *number of discussion contribution*, *number of wiki contributions*, *session duration time*, *number of activity days*, *average time for sessions*, *average time between sessions*, *number of inactivity consecutive days*, *time on task*, *average activity pace*, *number of page closed*, and *number of navigation link explored*. Thus, the first timeslot foresees 13 features, the second timeslot has 26 features (13 features coming from the previous timeslot and 13 new features), and so on. Truth labels are reported in column *dropout*. Furthermore, because of unbalanced classes (dropout and no-dropout), the dataset has been undersampled, taking 20k rows of non-dropout and 20k rows of dropout. It is important to note that the activities traced into the used dataset are all included in instructor-paced courses for which it was simple to decompose the timeline into 6 timeslots of fixed length. Moreover, the obtained dataset was further processed for value normalization (by means of the L2 norm) and discretization (5 bins of the same length for each column).

A. TRAINING AND TESTING

The training phase, as described before, is implemented as an iterative/incremental process over the dataset. The 80% of the dataset is used to train and test the individual classifiers. The remaining 20% of the dataset is used to test the whole approach.

1) ANALYSIS OF 3WD APPLICATION

Each iteration foresees the application of the 3WD algorithm (the thresholds have been empirically fixed as shown in Table 1 trying to hold stable CAR and CRR values for all iterations) and subsequently the construction of a three-way classifier by means of the CN2 Rule Induction algorithm. Such classifiers have been trained and tested by using a cross-validation approach. The sequential application of 3WD produces the results reported in Table 1 and Table 2.

TABLE 1. Parameters and results of the 6 iterations of 3WD (part I).

ITER	β	α	*GRAN. SIZE*	*#POS*	*#BND*	*#NEG*
#1	0.30	0.80	24.24	6024	20813	5163
#2	0.35	0.75	17.70	1655	16660	2498
#3	0.35	0.75	17.56	832	13835	1993
#4	0.30	0.75	15.94	914	11447	1474
#5	0.28	0.75	12.54	2631	7692	1124
#6	0.50	0.50	10.00	4428	0	3264

TABLE 2. Results of the 6 iterations of 3WD (part II).

ITER	*CAR*	*CRR*	*IAR*	*IRE*	*NNE*	*NPE*
#1	0.89	0.86	0.11	0.14	0.53	0.47
#2	0.90	0.87	0.10	0.13	0.52	0.48
#3	0.94	0.87	0.06	0.13	0.49	0.51
#4	0.86	0.89	0.14	0.11	0.47	0.53
#5	0.78	0.91	0.22	0.09	0.49	0.51
#6	0.65	0.69	0.35	0.31	0.00	0.00

[2] www.kddcup2015.com.

In particular, Table 1 shows the cardinalities of the three regions and mean granule size along the 6 iterations. It is interesting to see how the cardinality of region BND decreases along the timeslot sequence. This phenomenon is justified by the contraction of the universe of discourse (number of objects) along the sequence of iterations. As new iterations are executed, more and more objects are moved to regions POS (dropout) or NEG (non-dropout) from BND. In other words, the system improves its capability to decide as more knowledge is acquired and uncertainty decreases.

Table 2 reports the following analytics [22] for each iteration: CAR (Correct Acceptance Rate), CRR (Correct Rejection Rate), IAR (Incorrect Acceptance Rate), IRE (Incorrect Rejection Error), NNE (Non Negative Error) and NPE (Non Positive Error). CAR and CRR tending to 1.0 indicate an excellent performance in classifying objects as, respectively, POS region (certain dropout) and NEG region (certain non-dropout). IAR and IRE indicate errors in positioning objects in, respectively, POS region and NEG region. Hence, values tending to 0.0 are preferable. Furthermore, NNE and NPE indicate errors in BND region. It is clear that the sum of NNE and NPE is necessarily equal to 1.0. Therefore, a balance between such values is an acceptable result. Lastly, it is possible to conclude that sequential tri-partitioning helps to reduce uncertainty in data and can actively and positively support the work of the CN2 algorithm.

2) ANALYSIS OF CN2 APPLICATION

At each iteration, the objects, organized into three regions through 3WD, are used to train a classification model by means of CN2 Rule Induction algorithm. Such classification model acts as a three-way classifier and the three classes are: POS, BND and NEG. The idea is that if a new object is classified as BND, there is not sufficient information to predict a dropout or a non-dropout for such an object. Thus, the decision must be deferred and the next iteration classifier will try to provide a decision by using additional information coming new student's interactions. In order to train and test such a sequence of classifiers, the cross-validation (with 5 folds) approach has been used along the 6 iterations. The results of the train and test activity are reported in Table 3. Before commenting such a table it is interesting to note that the CN2 algorithm is configured to use entropy as an evaluation measure.

TABLE 3. Cross-validation results of CN2 algorithm.

ITER	*PREC.*	*REC.*	*ACC.*	*F1*	*SPEC.*	*AUC*
#1	0.99	0.99	0.99	0.99	0,99	0.99
#2	0.98	0.98	0.98	0.98	0,99	0.98
#3	0.98	0.98	0.98	0.98	0,96	0.98
#4	0.98	0.98	0.98	0.98	0,95	0.96
#5	0.97	0.97	0.97	0.97	0,93	0.98
#6	0.97	0.97	0.97	0.97	0,87	0.97

Table 3 shows that all classifiers provide good performance concerning accuracy, precision, recall, and F1-score. It is important to underline that the classifier of the last iteration predicts two classes (POS and NEG) and it is different from the other five classifiers predicting three classes (POS, BND and NEG). In this way, after the sixth iteration, a decision must be taken.

Furthermore, let's discuss the prediction performance of the whole process Such performance is measured by evaluating the integration of the 6 classifiers over the 20% of the dataset that is not used to train them. In particular, accuracy is 0.76, precision is 0.76, recall is 0.78 and F1-score is 0.77. The ROC curve is reported in Fig 5 and it produces an AUC of 0.87. It is important to underline that the the performance of the integrated pipeline is not a practical case given that the scores must be considered at each individual iteration to be compared to the state-of-the-art approaches. Such scores are reported here only for the aim of completeness and it could be considered a kind of worst case for our approach.

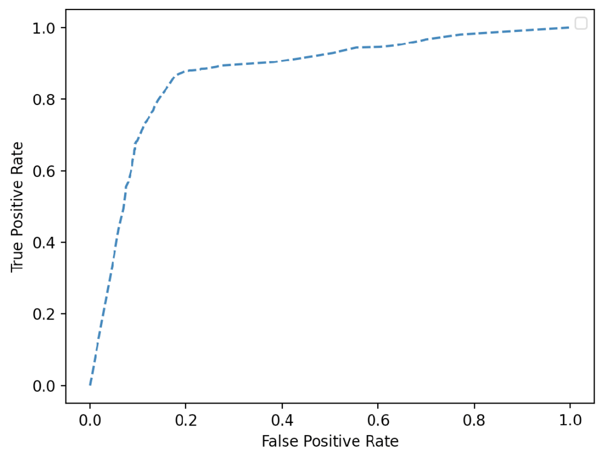


FIGURE 5. ROC-AUC of the whole process.

As emphasized before, one of the advantages of the proposed approach is its capability to explicitly characterize the different students' behaviors. In particular, it is possible to describe, for each iteration (at the end of each timeslot), three different behaviors (dropout-leading behavior, non-dropout behavior, uncertainty) by means of a set of IF-THEN rules. For instance, at iteration 2, the dropout-leading behavior is described by a set of rules containing the following ones:

- **IF** activity_days:1 $\in$ (0.133, 0.166] **THEN** REGION **is** POS
- **IF** inactivity_consecutive_days:1 $\in$ (0.255, 0.34] **AND** inactivity_consecutive_days:0 $\in$ (0.34, 0.425] **THEN** REGION **is** POS

As indicated in the sample rules, introduced before, the characterization of dropout-leading behaviors, at iteration 2, is described by using features constructed for both timeslot 0 and timeslot 1. Of course, the presented sample rules have

POS as consequent whilst the other two sets of rules have NEG and BND as consequents.

With respect to the adopted features, it has been applied the information gain (a well-known feature selection method) to rank their importance within different timeslots. In particular, *session duration time* and *average time for sessions* seem to be among the most important features along all the timeslots, while *average time between sessions* and *number of inactivity consecutive days* are more important during the first timeslots than the last ones. Time-related features (e.g., *session duration time*) seem to be more important than content-specific features (e.g., *number of videos played*, *number of accesses to the course material*). This aspect is reasonable given that the used data have been gathered from MOOCs dealing with very different subject matters and, consequently, requiring different studying methods. Lastly, the *number of videos played* seems to be the most important content-specific feature.

B. COMPARISON WITH OTHER WORKS

Let's compare the approach proposed in this paper (S3WDP) and those approaches, exploiting the same data, whose descriptions have been recognized from the scientific literature. The authors of [23] report a comparative analysis of the learning on KDD Cup 2015 Dataset (the dataset used to evaluate the approach proposed in the present work). The models evaluated in [23] have been tested by using cross-validation. The best results are obtained by using SVM (accuracy = 0.85, precision = 0.86, recall = 0.97, F1-score = 0.91). If considering all models (Random Forest, KNN, etc.) the accuracy goes from 0.78 to 0.85, precision goes from 0.78 to 0.86, recall goes from 0.97 to 1.0 and, lastly, F1-score goes from 0.88 to 0.91. Therefore, the results provided by S3WDP approach during the cross-validation step (see Tab.3) are better (or at least equal to) than the ones reported in the comparison paper.

Lastly, if considering the further evaluation of S3WDP (executing by using only data unexploited during the training phase), whose results are reported in Section VII-A2, it emerges that the proposed approach achieves acceptable results all the more so that it is able to be injected in a real-world educational decision support system and it can provide decisions incrementally, anticipating the decision when possible or deferring it when needed (lack of information).

Hence, the benefits of S3WDP approach are clear, given that it allows to early handle dropout-leading behaviors with worsening the performance of the classification task with respect to the state of the art (considering the same training data). Other excellent results (precision = 0.887, recall = 0.960, F1-score = 0.922, AUC = 0.872), using the same training data, are provided by the approach proposed in [14] that offers also the advantages of early decision-making. These results are obtained through the deployment of an approach similar to that of the present paper. It constructs a multidimensional time series to increase the number of features (and their informative density) as time goes on. In particular, the approach proposed in [14], namely, MFCN-VIB, obtains three models (10-days data, 20-days data and 30-days data). It is interesting to compare the performance of MFCN-VIB to those of S3WDP with respect to the early detection of a dropout-leading behavior. Thus, the MFCN-VIB 10-days data model and the first iteration (7 days) of S3WDP are considered. In particular, MFCN-VIB achieves precision = 0.832, recall = 0.937, F1-score = 0.881, AUC = 0.798. S3WDP achieves precision = 0.861, recall = 0.853, F1-score = 0.857, AUC = 0.875. The MFCN-VIB measures are obtained through a cross-validation evaluation and the S3WDP measures are obtained with training (80%) and testing (20%) data. Lastly, if considering cross-validation evaluation for S3WDP the results are reported in Tab. 3 and are all preferable to those of MFCN-VIB.

TABLE 4. Comparing different results on the same dataset (values are rounded).

	KDD CUP 2015	MFCN #1	MFCN #2	S3WDP #1	S3WDP #2	S3WDP #3
P	0.78-0.85	0.89	0.83	0.98	0.76	0.99
R	0.78-0.86	0.96	0.94	0.98	0.78	0.99
F	0.88-0.91	0.92	0.89	0.98	0.77	0.99
A	N/A	0.87	0.80	0.98	0.87	0.99

Such results are reported in Table 4, whose rows report precision (P), recall (R), F1-score (F), and AUC (A) for different approaches: best results in the competition (KDD CUP 2015), best results for MFCN-VIB (MFCN #1), results of 10-days model for MFCN-VIB (MFCN #2), mean values for the six iterations of the proposed method (S3WDP #1), results of the pipeline for integrated classifiers of the proposed method (S3WDP #2), results of the first iteration (7-days) of the proposed method (S3WDP #3). Such results clearly show the excellent performance of the proposed method if considering individual iterations that can be directly compared with the state-of-the-art approaches. Lastly, results in column S3WDP #2 must be considered for the aim of completeness. In this case, scores are lower because the classification error is the sum of the errors obtained during all iterations.

VIII. CONCLUSION AND FUTURE WORKS

The paper proposes a novel approach for defining an educational decision support system able to predict students' dropout on the basis of learning activities conducted by them through the usage of a MOOC platform. The paper adopts a reduced set of features extracted from the KDD Cup 2015 dataset. Thus, it can be generalized for many applications where data are produced by means of common logging operations (e.g., e-learning, e-commerce, virtual museum, etc.). The evaluation shows good results across multiple experimentation activities with respect to common metrics like precision, recall, accuracy, F1-score, and AUC. Moreover, several comparisons with similar approaches have been carried out with promising results. According to such comparisons, one of the main advantages of the proposed approach is its capability to be aware of course time and, consequently, support decisions iteratively as new information comes into the system. This capability is fundamental to rescue students from dropout when they start to show dropout-leading behavior. A further advantage is the capability to explicitly characterize the students' behavior by using the CN2 algorithm to produce a scrutable trained model. Among the other original aspects, the proposed approach includes a customized version of the Sequential Three-Way Decisions method that is used to reduce uncertainty and improve the prediction performance in order to support data streaming. A different technical approach supporting data streaming based on a Three-Way Decisions has been proposed in [24]. Lastly, in future works, the authors will focus on analyzing dropout patterns within specific subject areas in the context of MOOCs. The findings from this investigation will be detailed in upcoming research publications, shedding light on the critical issue of student attrition in online education.

ACKNOWLEDGMENT

The authors would like to thank the CODAS Laboratory[3] with DISA-MIS, University of Salerno, for the technological support.

REFERENCES

[1] F. Dalipi, A. S. Imran, and Z. Kastrati, "MOOC dropout prediction using machine learning techniques: Review and research challenges," in *Proc. IEEE Global Eng. Educ. Conf. (EDUCON)*, Apr. 2018, pp. 1007–1014.

[2] A. Roy, M. Yee, M. Perdue, J. Stein, A. Bell, R. Carter, and S. Miyagawa, "How COVID-19 affected computer science MOOC learner behavior and achievements: A demographic study," in *Proc. 9th ACM Conf. Learn. Scale*, 2022, pp. 345–349.

[3] J. Gardner and C. Brooks, "Student success prediction in MOOCs," *User Model. User-Adapted Interact.*, vol. 28, no. 2, pp. 127–203, May 2018.

[4] P. M. Moreno-Marcos, P. J. Muñoz-Merino, J. Maldonado-Mahauad, M. Pérez-Sanagustín, C. Alario-Hoyos, and C. D. Kloos, "Temporal analysis for dropout prediction using self-regulated learning strategies in self-paced MOOCs," *Comput. Educ.*, vol. 145, Feb. 2020, Art. no. 103728.

[5] Y. Zheng, Z. Gao, Y. Wang, and Q. Fu, "MOOC dropout prediction using FWTS-CNN model based on fused feature weighting and time series," *IEEE Access*, vol. 8, pp. 225324–225335, 2020.

[6] Z. Chi, S. Zhang, and L. Shi, "Analysis and prediction of MOOC learners' dropout behavior," *Appl. Sci.*, vol. 13, no. 2, p. 1068, Jan. 2023.

[7] W. Feng, J. Tang, and T. X. Liu, "Understanding dropouts in MOOCs," in *Proc. AAAI Conf. Artif. Intell.*, Jul. 2019, vol. 33, no. 1, pp. 517–524.

[8] P. Clark and R. Boswell, "Rule induction with CN2: Some recent improvements," in *Machine Learning—EWSL*. Porto, Portugal: Springer, 1991, pp. 151–163.

[9] Y. Yao, "Three-way decisions with probabilistic rough sets," *Inf. Sci.*, vol. 180, no. 3, pp. 341–353, Feb. 2010.

[10] C. Blundo, G. Fenza, G. Fuccio, V. Loia, and F. Orciuoli, "Sequential three-way decisions for reducing uncertainty in dropout prediction for online courses," in *Advanced Information Networking and Applications* (Lecture Notes in Networks and Systems), vol. 449. Cham, Switzerland: Springer, 2022.

[11] A. Hershkovitz and R. Nachmias, "Developing a log-based motivation measuring tool," in *Proc. EDM*, 2008, pp. 226–233.

[12] S. Ardchir, Y. Ouassit, S. Ounacer, H. Jihal, M. Y. E. Goumari, and M. Azouazi, "Improving prediction of MOOCs student dropout using a feature engineering approach," in *Advances in Intelligent Systems and Computing*. Berlin, Germany: Springer, 2020, pp. 146–156.

[13] W. Xing and D. Du, "Dropout prediction in MOOCs: Using deep learning for personalized intervention," *J. Educ. Comput. Res.*, vol. 57, no. 3, pp. 547–570, Mar. 2018.

[14] Z. Shou, P. Chen, H. Wen, J. Liu, and H. Zhang, "MOOC dropout prediction based on multidimensional time-series data," *Math. Problems Eng.*, vol. 2022, pp. 1–12, Apr. 2022.

[15] F. Pan, B. Huang, C. Zhang, X. Zhu, Z. Wu, M. Zhang, Y. Ji, Z. Ma, and Z. Li, "A survival analysis based volatility and sparsity modeling network for student dropout prediction," *PLoS ONE*, vol. 17, no. 5, May 2022, Art. no. e0267138.

[16] Y. Yao and X. Deng, "Sequential three-way decisions with probabilistic rough sets," in *Proc. IEEE 10th Int. Conf. Cognit. Informat. Cognit. Comput. (ICCI-CC)*, Aug. 2011, pp. 120–125.

[17] Z. Pawlak, "Rough sets," *Int. J. Comput. Inf. Sci.*, vol. 11, no. 5, pp. 341–356, Oct. 1982.

[18] Y. Yao, S. Greco, and R. Słowiński, "Probabilistic rough sets," in *Springer Handbook of Computational Intelligence*. Berlin, Germany: Springer, 2015, pp. 387–411.

[19] Y. Yao, "Three-way decisions and cognitive computing," *Cognit. Comput.*, vol. 8, no. 4, pp. 543–554, Aug. 2016.

[20] P. Clark and T. Niblett, "The CN2 induction algorithm," *Mach. Learn.*, vol. 3, no. 4, pp. 261–283, Mar. 1989.

[21] T. Zhang and B. Yuan, "Visualizing MOOC user behaviors: A case study on XuetangX," in *Intelligent Data Engineering and Automated Learning—IDEAL*. Springer, Sep. 2016, pp. 89–98.

[22] X. F. Deng, "Three-way classification models," Ph.D. dissertation, Dept. Comput. Sci., University of Regina, Regina, SK, Canada, 2015.

[23] S. Nithya and S. Umarani, "Comparative analysis of the learning on KDD cup 2015 dataset," *Webology*, vol. 19, no. 1, pp. 705–717, Jan. 2022.

[24] G. Fuccio, V. Loia, and F. Orciuoli, "Three-way decisions on streaming computing platforms supporting decision-making in complex large real-world environments," *IEEE Access*, vol. 10, pp. 122314–122328, 2022.

CARLO BLUNDO received the Ph.D. degree from the University of Napoli, in 1995. He is currently a Full Professor in computer science with Dipartimento di Scienze Aziendali-Management and Innovation Systems, University of Salerno. He has coauthored over 100 scientific publications. His main research interests include cryptography and data security.

VINCENZO LOIA (Senior Member, IEEE) received the master's degree in computer science from the University of Salerno, in 1985, and the Ph.D. degree in computer science from the University of Paris 6, in 1989. He is currently a Full Professor in computer science with the University of Salerno. He is also the Editor-in-Chief of *JAIHC* (Springer) and an associate editor of 14 international journals.

FRANCESCO ORCIUOLI (Member, IEEE) received the master's degree (cum laude) in computer science from the University of Salerno. He is currently an Associate Professor in computer science with the University of Salerno. He is the coauthor of more than 130 publications indexed by Scopus. His current research interests include data science and computational intelligence.

[3]https://www.disa.unisa.it/en/department/structures?id=404

Open Access funding provided by 'Università degli Studi di Salerno' within the CRUI CARE Agreement