Preprint of:
Nawaz, M. S., Nawaz, M. Z., Fournier-Viger, P., Luna, J.-M. (2024). Analysis and Classification of Employee Attrition and Absenteeism in Industry: A Sequential Pattern Mining-Based Methodology. Computers in Industry (COMIND). Elsevier, August 2024, article 104106.
DOI: 10.1016/j.compind.2024.104106

Analysis and Classification of Employee Attrition and Absenteeism in Industry: A Sequential Pattern Mining-Based Methodology

Date of receiving / date of acceptance

Abstract Employee attrition and absenteeism are major problems that affect many industries and organizations, resulting in diminished productivity, elevated costs, and losses. These phenomena can be attributed to multiple factors that are difficult to anticipate for human resources or management. Therefore, this paper proposes a content-based methodology for the analysis and classification of employee attrition and absenteeism that can be used for talent analysis and management, a task that is traditionally carried out ex-post. The developed methodology, called E(3A)CSPM, is based on SPM (sequential pattern mining). In the methodology, four public datasets with diversified employee data are adopted, which are initially transformed into a suitable format. Then, SPM algorithms are applied to the transformed datasets to reveal recurring patterns and rules of features. The discovered patterns and rules not only offer information regarding features that have a key role in employee attrition and absenteeism but also their values. These frequent patterns of features are thereafter used to classify/predict employee attrition and absenteeism. Eight classifiers and multiple evaluation metrics are used in experiments. The performance of E(3A)CSPM is contrasted with state-of-the-art approaches for employee attrition and absenteeism and the obtained findings reveal that E(3A)CSPM surpasses these approaches.

Keywords Employee Attrition · Absenteeism · Classification · Sequential pattern Mining · Analysis.

1 Introduction

Employee attrition refers to the reduction (leaving) of personnel in any industry/organization [1]. Attrition can occur due to unpredictable or uncontrollable factors, such as resignation, voluntary retirement, termination, long-term illness, structural changes, and layoffs [2–4]. On the other hand, absenteeism at work generally refers

to an employee's habitual pattern of absence from his or her duty or obligation [5]. Employee absenteeism can generate problems in the employee-employer relationship and has an impact on an industry's productivity [5–8]. According to the JOLTS (Job Openings and Labor Turnover Survey), 3.7 million employees quit their jobs in September 2023, representing 2.3% of the workforce [9]. Replacing an employee costs almost three to four times the salary of the position [10]. Moreover, the US Bureau of Labor Statistics reported an average employee absence rate of 3.6% in 2022 [11].

Employee attrition and absenteeism are a growing problem in industries/organizations as they directly affect their performance and culture, disrupt ongoing tasks, and create additional costs for re-employment, re-training and influence long-term growth strategies [1, 3, 12]. In the past, industries relied on traditional methods, such as interviews, surveys and post-action response to obtain data about this problem [2]. However, such methods are often unable to get accurate and honest answers from employees and the bias in the obtained data can lead to errors. With the recent advances in machine learning (ML), industries and organizations can not only take preemptive action by predicting the various reasons and factors for employee attrition and absenteeism, but also improve their policies, culture and regulatory environment to help retain employees.

Literature on computational research for the employee attrition and absenteeism problems can be divided into three broad groups: (1) ML-based methods [13–16], (2) Deep learning (DL)-based methods [5, 7, 17–19] and (3) ensemble-based methods [1, 3, 20–23] (for more details, see Section 2). These studies used various feature engineering and encoding methods for classification and prediction. However, the majority of methods have issues with computing efficiency, interpretability, and scalability. Moreover, the generalization ability of those models remains an open question as they have been evaluated on few datasets, generally only one (as it will be observed in Section 2). To analyze and manage employee data, simple yet sophisticated and intelligible techniques, such as frequent pattern mining (FPM) (also sometimes called association analysis) [24], are desirable because attrition and absenteeism depend on various causes and factors, and the data generally has atypical properties. FPM is a set of techniques from data mining that aims at identifying interesting patterns in data that can capture important information from the data, where patterns can have various forms such as rules and be interpretable as they directly refer to attribute values from the data.

In this paper, we aim to test the hypothesis that FPM techniques could be useful for employee data analysis and management, and more specifically that the discovered patterns could be used as features for the effective classification for employee attrition and absenteeism. In particular, sequential pattern mining (SPM) [25] techniques are employed, which are efficient methods for analyzing sequential data based on sequential frequent patterns and sequential rules. Frequent sequential patterns have a support (occurrence frequency) that is no less than a user-specified parameter called minimum support (*minsup*). Frequent sequential rules have a support that is no less than a *minsup* and a minimum confidence or probability (*minconf*) parameter, also set by the user. SPM has been used recently in many applications, including to extract hypernym relations from texts [26, 27], genome analysis [28, 29], tourist movement

analysis [30], market basket analysis [31], malware behavior analysis [32], workload prediction in cloud environment [33], mining discrete clinical data [34] and for the analysis of protein structure sequences [35]. To the best of our knowledge, no study is present in the literature that show how frequent sequential patterns of employee attributes (features) can be used for the efficient classification of employee attrition and absenteeism. The proposed SPM-based employee attrition and absenteeism approach is evaluated on multiple datasets to gain insights into its effectiveness and generalization ability across various data sources and characteristics. The presented approach is embedded in a new methodology for analyzing employee data, called E(3A)CSPM (Employee Attrition and Absenteeism Analysis and Classification using Sequential Pattern Mining). It provides a pipeline with data transformation, sequential pattern extraction and classifier training, offering:

- A SPM-based approach for employee attrition and absenteeism analysis. This approach first transforms employee datasets into a format that is well-suited for SPM. The resulting datasets are then fed to SPM algorithms to discover the frequent sets of features and their values, and the strong sequential relationships among them in the form of sequential patterns and rules.
- An approach for detecting employee attrition and absenteeism that uses the frequent patterns, extracted by SPM algorithms, as features. In particular, eight classifiers are employed for classification using the patterns and various evaluation metrics are used to compare their performance.

Extensive experiments were done with multiple assessment criteria to verify the efficiency of this approach, using four different datasets of employee attrition and absenteeism. The experimental results reveal that using the E(3A)CSPM methodology to find frequent sequential patterns of features in attrition and absenteeism data and using discovered patterns leads to improved classification performance as compared to using all of the features. Overall, decision tree (DT) performed well for both binary and multi-class classification. Moreover, it was found that the designed E(3A)CSPM methodology outperforms recently developed approaches for employee attrition and absenteeism detection. Through the frequent patterns found in this study, some valuable insights can be obtained into the causes/factors (features) that play a major role in employee attrition and absenteeism. This contributes to a deeper understanding of the characteristics and commonalities of employee data, potentially aiding in the development of more robust detection models and strategies. The current research has the potential to support industries in conducting quick, automatic, and well-informed analysis, extract important information (key features) particularly in employee data where ordering is important, and build essential knowledge bases, which could help control or reduce attrition and absenteeism.

The rest of this paper is divided into five sections: Section 2 gives an overview of ML, DL and ensemble-based methods for analyzing and detecting employee attrition and absenteeism. In Section 3, the datasets used in this work are described. The proposed E(3A)CSPM methodology is then presented in Section 4, which is used for employee attrition and absenteeism analysis and classification. Section 5 presents and discusses experimental results. Furthermore, the performance of E(3A)CSPM is

compared to that of other recent approaches. Finally, in Section 6, a conclusion is offered.

2 Related Work

This section reviews the recent computational methods for analyzing and detecting employee attrition and absenteeism published in the last four years (2020-2023).

Table 1: Summary of related works that use ML, DL and ensemble learning algorithms to address the employee attrition and absenteeism problem. FSM: Feature selection model(s), FA: Feature analysis

Ref.	Dataset	FSM	FA	Classifier(s)	Evaluation Metrics
[1]	IBM-1	PCA	No	DT, LR, FR, GB, AB, Ensembling	ACC, P, R, F1, AUC
[2]	IBM-1	Chi-Square	Yes	RF, MLP, XGboost	ACC, P, R, F1
[3]	IBM-1	–	Yes	SVM, LR, ANN, XGB, RF, Stacking-based ensemble model	ACC, P, R, F1, AUC
[5]	BCC-3	PCA	Yes	Deep NN, DT, RF, SVM	ACC, P, R, F1, ROC
[6]	BCC-3	CFS	Yes	ZeroR, NB, 3 kNNs, J48	ACC, P, R, F1, ROC
[7]	BCC-3	LRP	Yes	MLP	ACC, R, S
[8]	BCC-3	–	Yes	LR, DT, RF, AdaBoost (AB)	RMSE
[13]	Selfmade	–	Yes	SVM, DT, RF, GNB, LR, kNN	ACC, F1
[14]	IBM-1	Correlation Matrix	Yes	RF, DT, LR	ACC, P, R, F1
[15]	IBM-1	Pearson Correlation	Yes	kNN, MLP, LR	ACC
[16]	IBM-1	Correlation Matrix	Yes	kNN, RF	ACC, P, R, F1, AUC, FPR, S
[17]	IBM-1, HRA-4, Selfmade	Recursive Feature Elimination, SelectKBest	Yes	DT, SVM, LR, DNN, LSTM, CNN, RF, XGB, Stacked-ANN, Voting classifier	ACC, F1
[18]	IBM-1	Pearson Correlation	No	ANN+SVMSmote	ACC, P, R, F1
[19]	IBM-1, HRA-4, Selfmade	Permutation Importance	Yes	RF, SVM, LDA, kNN, Bagging, AB, XGB, LR, NB, Deep RF, NN	ACC, P, R, F1, AUC, S
[20]	IBM-1, HRA-4	Low Correlation with the Target	Yes	15 classifiers and Ensembling	ACC, P, R, F1, AUC
[21]	SAS	–	No	RF, GB, MLP, Ensembling	ACC, Lift, R, F1
[22]	IBM-1	Pearson Correlation	Yes	LR, CT, RF, NN, NB, Ensembling	ACC, F1, AUC
[23]	Selfmade	IG	Yes	GB, RF, NN, kNN, SVM, NB, LR, Ensembling	ACC, P, R, F1, ROC
[36]	IBM-1	PCA	Yes	RF, GB	ACC, P, R, F1, Support
[37]	IBM-1	–	Yes	SVM, LR, RF, XGBoost	AUC
[38]	IBM-1	EEDA	Yes	ETC, SVM, LR, DT	ACC, P, R, F1, ROC
[39]	IBM-1	Correlation Matrix	Yes	GNB, BNB, DT, LR, RF, MNB	ACC, P, R, F1
[40]	IBM-1	Chi-Square	Yes	LR, DT, RF, NB, kNN, SVM	ACC, P, R, F1, AUC
[41]	HRA-4	KPCA	Yes	NB, LR, KPCA+AdpKmeans	ACC, P, R, AUC
[42]	IBM-1	LIME, SHAP	Yes	LightGBM	AUC
[43]	IBM-1	Correlation Matrix	Yes	LR, kNN, RF	ACC, P, R, F1
[44]	IBM-1	Correlation Matrix	Yes	GNB, BNB, LR, kNN, DT, RF, SVM, linear SVM	ACC, P, R, S, F1
[45]	IBM-1	Max-out	Yes	LR, kNN, RF, FT, NB	ACC, P, R, F1
[46]	BCC-3	RFS, CFS, IGFS	Yes	NB, LR, MLP, kNN, Bagging, J48, RF	ACC, P, R
[47]	BCC-3	–	Yes	MLR, Tree, kNN	ACC, P, R, F1
[48]	IBM-1	–	No	LR, 3 SVMs, Boosted Trees	ACC
[49]	IBM-1	SHAP	Yes	LR, DT, RF, AB, kNN, XGBoost	ACC, P, R, F1, RMSE
[50]	IBM-1	Correlation Matrix	Yes	LR, kNN, DT, RF, AdaBoost	ACC, P, R, F1, AUC
[51]	IBM-1	MRMA, Chi-Square, ANOVA, Kruskal-Wallis	Yes	SVM, XGBoost, LR, DT, NB	ACC, P, R, F1
[52]	ORACLE ERP	Intensive Optimized PCA	No	RF	ACC, P, R, F1, ROC
[53]	HRA-4	Fisher score, Chi-Square, Spearman Correlation, and R Coefficient Correlation	Yes	SVM, DT, NN, LR, DF	ACC, P, R

In the literature, the majority of the studies [2, 6, 8, 13–16, 36–52] used popular ML classifiers, and studies [5, 7, 17–19] and [1, 3, 20–23] used ML as well as DL and

ensembling for employee attrition, turnover and absenteeism classification/prediction. The studies [1–3, 14–20, 22, 36–40, 42–45, 48–51] used the IBM (called IBM-1) dataset and [17, 19, 20, 41, 53] used Kaggle HR (called HRA-4) dataset. For employee absenteeism classification/prediction, [5–8, 46, 47] used the courier company dataset from Brazil, called BCC-3. The studies used different testing training ratios, or k-fold cross validation, to analyze classifiers, and their performance was measured with different metrics, of which Accuracy (ACC), Precision (P), Recall (R) and F1-score (F1) were the most common. All the studies used various feature engineering, encoding and selection methods, of which principal component analysis (PCA) and correlation matrix heatmap were the most common. Table 1 lists recent and relevant studies and compares them in terms of the above characteristics.

The studies [2, 3, 5–8, 13–17, 19, 20, 22, 23, 36–47, 49–51, 53] used various statistical, descriptive, and exploratory techniques to analyze features in the dataset and their role in employees quitting the organization or their role in their absenteeism. For example, Chung et al. [3] identified that three features (relationship satisfaction, overtime and environmental satisfaction) were significant contributors to attrition. Skorikov et al. [6] performed three experiments on the prediction of absenteeism. In the first experiment, four features (the month of absence, age, disciplinary failure, social drinker), found by the using the CFS method, were used. The second experiment used all the features from the BCC-3 dataset. The third experiment was conducted with only one feature (disciplinary failure). Atef et al. [16] used five features (salary, distance from home, marital status, age, and gender) from the IBM-1 dataset. Raza et al. [38] found the significance of factors such as age, hourly rate, monthly income and job level. Bansal et al. [40] used 20 features, according to the Chi-Square test, and ignored 15 features as they were unable to contribute effectively in the target prediction of "Attrition". Fallucchi et al. [44] examined how objective factors in the IBM-1 dataset influence employee attrition. Naz et al. [53] found top features in the HRA-4 dataset and used them in the classification process.

Sampling techniques such as SMOTE [54] were used in [3, 5, 6, 18, 19, 38, 50] to overcome the class imbalance problem in the dataset(s). Only two studies [46, 48] discussed the training time and total time for the classifiers while detecting employee absenteeism and attrition, respectively. The studies discussed in this section used various feature engineering, encoding, and selection methods for classification/prediction. However, because the proposed methods are tested on limited datasets, usually one, they have issues with efficiency, interpretability, scalability, and generalization. Moreover, some of the feature extraction methods are expensive. Differently from prior work, this study extracts sequential frequent patterns of features that can be used for reliable classification and detection purposes.

3 Datasets

In this study, four publicly available datasets for employee attrition and absenteeism from Kaggle are used to access the performance and efficacy of the proposed methodology. The first dataset is the IBM HR Analytics Employee Attrition & Performance[1],

[1] kaggle.com/datasets/pavansubhasht/ibm-hr-analytics-attrition-dataset

referred to as IBM-1, containing 35 features and 1,470 samples. In IBM-1, 1,233 employees were from the *No* attrition group, whereas the remaining 237 employees belonged to the *Yes* attrition group. We removed three features from IBM-1: *Employee count*, since the numbers are all sequential (1, 2, 3, etc.) for this feature; *Over18*, because it is *Yes* for every employee; and *Standard hours*, because all employees had the same standard hours (80). Thus, the IBM-1 dataset in this work has 32 features, of which 24 are numerical and 8 are categorical.

Table 2: Statistics about the four datasets

Dataset	Samples	Features (Numerical/Categorical)	Dependent Feature
IBM-1	1,470	32 (24/8)	Attrition (237 Yes/1,233 No)
HRD-2	311	28 (19/9)	Terminated (104 Yes/207 No)
BCC-3	740	20 (20/0)	Absenteeism Time (740)
HRA-4	14,999	11 (7/4)	Left (3,571 Yes/11,428 No)

The second dataset, created by Rich Huebner and Carla Patalano, is the Human Resources Data Set[2], here called HRD-2. It contains 36 features and 311 samples. Two features *Employee_Name* and *Employee ID* (*EmpID*) are not considered in this study as they are unimportant features for analysis and classification. Five features *Position*, *Sex*, *Martial Status (MaritalDesc)*, *Manager name* and *Employment Status* are duplicate of *Position ID*, *Gender ID*, *Marital Status ID*, *Manager ID* and *TermD* respectively. The feature *Date of termination* is also not considered as most of the employees are active and this features is empty for them. Thus, the HDR-2 dataset in this work has 28 features, in which 19 are numerical and 9 are categorical.

The third dataset consists of records about employee absenteeism, containing 21 numerical features and 740 samples, at a courier company in Brazil from July 2007 to July 2010[3], called BCC-3. Only one feature *ID* was discarded from this dataset. The fourth dataset is the HR Analytics data set[4], referred to as HRA-4. It contains 14,999 samples, of which 3,571 belonged to employees that left the company and the remaining 11,428 belonged to employees that have not left. HRA-4 dataset in this work has 11 features, in which 7 are numerical and 4 are categorical. Statistics about the four datasets and feature details are given in Table 2 and Table 3 respectively. The dependent feature in each dataset is in bold in Table 3. Many samples from two datasets (BCC-3 and HRA-4) have missing values for some features. MICE (Multiple Imputation by Chained Equation) statistical method [55] was used to fill the missing values in these two datasets.

In BCC-3, the feature *absenteeism time in hours* is considered the dependent (class) feature. This attribute contains values that are continuous. However, as suggested in previous work [6, 47], classification of absenteeism time in terms of categories allows the model to classify/predict different degrees of absence on test data. We used the same categories (classes) for this feature (shown in Table 4).

[2] kaggle.com/datasets/rhuebner/human-resources-data-set. rpubs.com/rhuebner/HRCodebook-13

[3] www.kaggle.com/datasets/tonypriyanka2913/employee-absenteeism

[4] kaggle.com/datasets/cezarschroeder/human-resource-analytics-dataset

Table 3: Feature names in the four datasets

Dataset 1 (IBM-1)
(1) Age, (2) **Attrition**, (3) BusinessTravel, (4) DailyRate, (5) Department, (6) DistanceFromHome, (7) Education, (8) EducationField, (9) EmployeeNumber, (10) EnvironmentSatisfaction, (11) Gender, (12) HourlyRate, (13) JobInvolvement, (14) JobLevel, (15) JobRole, (16) JobSatisfaction, (17) MaritalStatus, (18) MonthlyIncome, (19) MonthlyRate, (20) NumberofCompaniesWorked, (21) OverTime, (22) PercentSalaryHike, (23) PerformanceRating, (24) RelationshipSatisfaction, (25) StockOptionLevel, (26) TotalWorkingYears, (27) TrainingTimesLastYear, (28) WorkLifeBalance, (29) YearsAtCompany, (30) YearsInCurrentRole, (31) YearsSinceLastPromotion, (32) YearsWithCurrentManager
Dataset 2 (HRD-2)
(1) MarriedID, (2) MaritalStatusID, (3) GenderID, (4) EmpStatusID, (5) DepartmentID, (6) PerfScoreID, (7) FromDiversityJobFairID, (8) Salary, (9) **Termd**, (10) PositionID, (11) State, (12) Zip, (13) DOB, (14) CitizenDesc, (15) HispanicLatino, (16) RaceDescription, (17) DateOfHire, (18) TerminationReason, (19) Department, (20) ManagerID, (21) RecruitmentSource, (22) PerformanceScore, (23) EngagementSurvey, (24) EmployeeSatisfaction, (25) SpecialProjectsCount, (26) LastPerformanceReviewDate, (27) DaysLateInTheLast30Days, (28) Absences
Dataset 3 (BCC-3)
(1) ReasonForAbsence, (2) MonthofAbsence, (3) DayofTheWeek, (4) Seasons, (5) TransportationExpense, (6) DistanceFromResidenceToWork, (7) ServiceTime, (8) Age, (9) WorkLoadAverage/Day, (10) HitTarget, (11) DisciplinaryFailure, (12) Education, (13) Son, (14) SocialDrinker, (15) SocialSmoker, (16) Pet, (17) Weight, (18) Height, (19) BodyMassIndex, (20) **AbsenteeismTimeInHours**
Dataset 4 (HRA-4)
(1) SatisfactionLevel, (2) LastEvaluation, (3) NumberofProjects, (4) AverageMonthlyHours, (5) TimeSpentInCompany, (6) WorkAccident, (7) **Left**, (8) PromotionInLast5Years, (9) Department, (10) Salary, (11) Smoker

Table 4: Categories for the *absenteeism time in hours* feature in BCC-3

Absent time in hours	Class	Samples
0	A	36
1-15	B	642
16-120	C	62

4 Methodology

The proposed E(3A)CSPM methodology (depicted in Figure 1) for the analysis and detection/classification of employee attrition and absenteeism consists of four main parts:

1. *Datasets pre-processing and abstraction*: The first step is to pre-process the datasets so that they are ready for applying SPM. This is accomplished by transforming each sequence into a discrete sequence. In this format, a positive integer represents each feature from the original datasets.
2. *Learning via SPM*: The second step is to apply various algorithms to extract different types of sequential patterns from the abstracted datasets to uncover features that are frequently correlated. The result is frequent patterns representing sequential relationships among the discovered frequent features.
3. *Classification via frequently occurring features*: The third step is to classify/detect employee attrition and absenteeism using the frequent patterns of features and

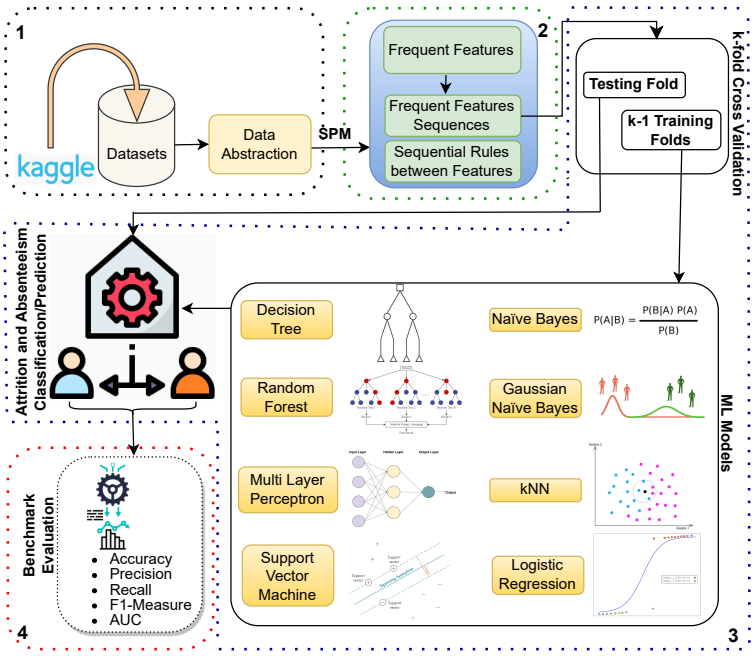


Fig. 1: The E(3A)CSPM methodology for the analysis and classification of employee attrition and absenteeism. E(3A)CSPM has four steps: (1) Pre-processing and abstracting the datasets, (2) Learning sequential patterns and rules using SPM, (3) Classifying employees using the discovered patterns of features and their values in the datasets by training different classifiers, and (4) Evaluating the methodology through extensive experiments.

their values, discovered in Step 2. Several classifiers are utilized, and different metrics are used to assess their performance.

4. Evaluation: Finally, experiments are preformed to evaluate the performance of E(3A)CSPM and compare it with state-of-the-art employee attrition and absenteeism approaches.

The following subsections provides more detail about the first three steps of E(3A)CSPM.

4.1 Datasets Pre-processing and Abstraction

The first step is data pre-processing, which consists of transforming features of each dataset into a specific integer-based format. We discovered that in the datasets, different features can have the same value. To avoid any ambiguity, each value of different features in the datasets is encoded into a distinct positive integer, that are called feature values in the remaining of the paper. The next paragraphs discuss key concepts related to sequences.

Given a dataset, let $F = \{F_1, F_2, ..., F_m\}$ denote the set of its feature values. Then, any subset FS of feature values ($FS \subseteq F$) is called a *feature values set*. The number of feature values (set cardinality) in a feature values set FS is represented by the notation $|FS|$. If FS contains k feature values (i.e. $|FS| = k$), we say that FS has a length of k. In addition, we call FS a *k-FS*. For example, the features set {*Age, Attrition, BusinessTravel, DailyRate, Department*} from the IBM-1 dataset has five features, and hence a length of five. To allow a systematic exploration of the search space of patterns, a relation $\prec$ is further defined on the set F of all feature values. This order is a total order that is internally used by SPM algorithms to guide the search for patterns and avoid finding duplicate patterns [25]. In practice, any total order can be used, and thus, the lexicographical order is used in the implementation of E(3A)CSPM.

Based on the concept of feature values set, a features sequence is a list $S = \langle FS_1, FS_2, ..., FS_n \rangle$ of feature values sets that are sequentially ordered (where $FS_i \subseteq F$ for $1 \leq i \leq n$). A *features values corpus* dataset FCD is a list of multiple features sequences. The notation $FCD = \langle S_1, S_2, ..., S_p \rangle$ denotes a features corpus dataset with p sequences. It is assumed that each sequence has a unique identifier (ID), which is $1, 2, ..., p$ in this notation. For example, Table 5(a) shows a FCD containing the first seven features of IBM-1 for employees represented by the IDs 1, 2, 3, and 4. To apply SPM, the raw data must be formatted in a particular way. More precisely, the features and their sequences of values in a FCD are transformed into sequences of integers. In that format, each positive integer represents a value of a distinct feature. Therefore, values that are identical in the original dataset for two different features are denoted using a different integer in the transformed dataset to differentiate them. The result is an abstracted dataset, where each line represents the feature sequence of an employee and is a list of integers. Moreover, two special codes are used in that format, namely -1 to separate feature values, an -2 to indicate the end of a sequence. For instance, the four feature sequences of Table 5(a) are converted into the integer feature sequences shown in Table 5(b).

Table 5: (a) A sample of *FCD* and (b) Features and their values as positive integers

(a)

ID	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education....
1	41	Yes	Travel_Rarely	1102	Sales	1	2 ....
2	49	No	Travel_Frequently	279	R & D	8	1 ....
3	22	No	Non_Travel	1123	R & D	16	2 ....
4	46	No	Travel_Rarely	945	HR	5	2 ....

(b)

ID	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education ....
1	91141	9121	9131	9141102	9151	9161	9172 ....
2	91149	9120	9132	914279	9152	9168	9171 ....
3	91122	9120	9133	9141123	9152	91616	9172 ....
4	91146	9120	9131	914945	9153	9165	9172 ....

4.2 Learning using SPM

In the second step, a FCD is analyzed to identify frequent patterns in its features sequences. This is done by searching for subsequences that occur with a high frequency (have many occurrences). Formally, assume that we have two features sequences S_a and S_b that are defined as $S_a = \langle a_1, a_2, ..., a_p \rangle$ and $S_b = \langle b_1, b_2, ..., b_q \rangle$. Then, S_a is called a *subsequence* of S_b if an only if there exist integers $1 \leq i_1 < i_2 < ... < i_p \leq q$, such that $a_1 \subseteq b_{i1}, a_2 \subseteq b_{i2}, ..., a_p \subseteq b_{iq}$. In that case, it is also said that S_a is present in S_b, and this relationship is written as $S_a \sqsubseteq S_b$. For instance, the sequence $\langle \{Yes\}, \{TravelRarely\} \rangle$ is a subsequence of sequence $\langle \{Yes\}, \{TravelRarely\}, \{Sales\} \rangle$. Thus, $\langle \{Yes\}, \{TravelRarely\} \rangle \sqsubseteq \langle \{Yes\}, \{TravelRarely\}, \{Sales\} \rangle$.

In SPM, a subsequence importance and interestingness can be accessed by various measures, the most common of which is the *support* measure. In a FCD, the number of sequences (S) containing S_a is the *support* of S_a, which is denoted by the symbol $sup(S_a)$:

$$sup(S_a) = |\{S | S_a \sqsubseteq S \wedge S \in FCD\}|$$

The aim of SPM is to do the complete enumeration of all the *frequent subsequences* in a sequential dataset (here, a FCD). A sequence S is said to be a *frequent sequence* (also known as a *sequential pattern*) iff $sup(S) \geq minsup$, where $minsup$ (minimum support) is a cut-off threshold selected by the user. This is a difficult problem to solve since there are numerous possible patterns. For instance, if there are n items in a sequence (feature values in this work), then up to $2^n - 1$ distinct subsequences may have to be considered. For most datasets, finding the support of every possible subsequence using a naive approach is not possible. However, over the last two decades, a number of effective algorithms have been created that can detect all possible sequential patterns without having to search through all potential subsequences.

Algorithms for SPM comb through the search space of sequential patterns by performing two operations: *s-extensions* and *i-extensions*. A sequence $S_a = \langle a_1, a_2, .. ., a_n \rangle$ is a *prefix* of another sequence $S_b = \langle b_1, b_2, ..., b_m \rangle$, if $n < m$, a_1 = b_1, a_2 = b_2 , ..., a_{n-1} = b_{n-1}, where a_n is equal to the first $|a_n|$ items of b_n according to the $\prec$ order. SPM algorithms ensure that the same potential patterns are not visited more than once during the searching process by following a specific order, $\prec$. Note that the final outcome (sequential patterns) produced by SPM algorithms is unaffected by the selection of $\prec$. For an time x, S_b is said to be an *s-extension* of S_a if S_b = $\langle a_1, a_2, ..., a_n, \{x\} \rangle$. Similarly, S_c is said to be an *i-extension* of S_a, for an item x, if S_c = $\langle a_1, a_2, ..., a_n \cup \{x\} \rangle$. SPM techniques typically use either a depth-first search or a breadth-first search with a variety of data structures and optimizations.

A SPM algorithm used in this work is TKS (Top-k Sequential) [56]. TKS searches a database for the top-*k* most frequent sequential patterns, where k is a user-specified input value. The value of the parameter *k* denotes the number of patterns to be discovered. The main reason for using TKS is that it lets us directly specify the number of sequential patterns to discover via the parameter *k*, unlike traditional SPM algorithms

that require setting a minimum frequency threshold. Using the parameter *k* provides the convenience of knowing exactly how many patterns will be output before running the TKS algorithm and thus eliminate the need of running the algorithm several times to get a desired number of patterns (e.g. by setting *k* = 500, TKS will find the top 500 most frequent patterns). To find the desired patterns, TKS uses a candidate generation process that follows a depth-first search and a vertical database representation. That database representation enables TKS to count patterns without scanning the database, which improves its performance on dense or long sequences. TKS also employs other strategies to reduce the search space, such as the PMAP (Precedence Map) data structure and a bit vector representation with an efficient join operation. For more details about TKS, please refer to [56].

One of the main drawbacks of traditional SPM algorithms such as TKS is that they may discover too many sequential patterns, most of which are not interesting or important for users. Sequential patterns with high support but low confidence are not useful in decision-making or prediction/forecasting tasks. Therefore, another type of patterns, known as sequential rule, can be discovered [57]. A sequential rule represents a relationship between two sets of items (feature values in this work) that considers both the support and the confidence (conditional probability) of the items. In this work, a sequential rule $Y \rightarrow Z$ represents a relationship between two feature value sets $Y, Z \subseteq FS$ s.t. $Y \cap Z = \emptyset$ and $Y, Z \neq \emptyset$. A sequential rule has the form $r : Y \rightarrow Z$ and means that in a sequence, Y's items (feature values) will be likely followed by Z's items. A sequence $S_a = \langle a_1, a_2, .., a_n \rangle$ contains a feature value set Y ($Y \sqsubseteq S_a$) iff $Y \subseteq \bigcup_{i=1}^{n} a_i$. Moreover, S_a contains the rule r ($r \sqsubseteq S_a$) iff there exists an integer k s.t. $1 \leq k < n$, $Y \subseteq \bigcup_{i=1}^{k} a_i$ and $Z \subseteq \cup_{i=k+1}^{n} a_i$. The confidence and support of the rule r in FCD is defined as:

$$conf_{FCD}(r) = \frac{|\{S | r \sqsubseteq S \wedge S \in FCD\}|}{|\{S | X \sqsubseteq S \wedge S \in FCD\}|}$$

$$sup_{FCD}(r) = \frac{|\{S | r \sqsubseteq S \wedge S \in FCD\}|}{|FCD|}$$

A rule r is a *frequent sequential rule* iff $sup_{FCD}(r) \geq minsup$ and r is a *valid sequential rule* iff it is frequent and $conf_{FCD}(r) \geq minconf$, where the thresholds $minsup$, $minconf \in [0, 1]$ are set by the user. Sequential rule mining in a dataset deals with enumerating all the valid sequential rules. We use the ERMiner (Equivalence class based sequential Rule Miner) algorithm [57] in this work to find frequent sequential rules in a feature corpus dataset. ERMiner uses a vertical database representation and uses equivalence classes of rules having the same antecedent and consequent to explore the entire search space of rules. To further explore the search space of frequent sequential rules, it uses left and right merges operations. The Sparse Count Matrix (SCM) technique is used for search space pruning that enables ERMiner to perform more efficiently than earlier algorithms for mining sequential rules. In summary, SPM algorithms differ from each other based on (1) the use of a depth-first or breadth-first search, (2) the use of a vertical or horizontal database representation and specific data structures, and (3) the support measure that is used to count patterns in datasets and output those that satisfy the user-specified constraints.

4.3 Classification

The third step is to use the frequent patterns of features and their values discovered with SPM to classify employee attrition and absenteeism. The datasets used for attrition and absenteeism classification contain employee records (sequences) of various lengths. A careful examination of the FCD datasets revealed that many sequences (pertaining to both attrition and absenteeism instances) contain feature occurrences that appear multiple times consecutively. In the context of employee-related records, these redundant occurrences of features do not provide much helpful information for classification. Hence, during the classification process, repeated features are considered a single entity.

More precisely, E(3A)CSPM utilizes the sequential frequent patterns discovered by SPM algorithms for the classification of attrition and absenteeism datasets. Two types of classification are carried out: binary and multi-class (MC). Binary classification is utilized for attrition datasets (IBM-1, HRD-2 and HRA-4) and MC classification is used for the absenteeism dataset (BCC-3). For a selected dataset, binary classification assigns "yes" or "no" labels to each sequence (employee record) representing whether they belong to that class or not.

For both binary and MC classification, seven standard ML algorithms and one DL algorithm are used, which are: (1) Multinomial Naive Bayes (MNB), (2) Gaussian Naive Bayes (GNB), (3) Decision Tree (DT), (4) Random Forest (RF), (5) Multilayer Perceptron (MLP), (6) Support Vector Machine (SVM), (7) k-Nearest Neighbors (kNN) and (8) Logistic Regression (LR) [58, 59]. A brief description of the classification models is given next.

MNB is a probabilistic classifier based on Bayes' theorem. Despite its "naive" assumption of feature independence, it often performs well and is computationally efficient. The probability of a class C_i given some input features $P(C_i|x_1, x_2, ..., x_n)$, where $P(C_i)$ is the prior probability of class C_i, is calculated by using Bayes' theorem (Equation 1).

$$P(C_i|x_1, x_2, ..., x_n) = \frac{P(C_i) \cdot P(x_1|C_i) \cdot P(x_2|C_i) \cdot ... \cdot P(x_n|C_i)}{P(x_1) \cdot P(x_2) \cdot ... \cdot P(x_n)} \tag{1}$$

GNB is another variant of NBs designed for continuous data. It assumes that features follow a Gaussian distribution and is commonly used in classification tasks where features are real-valued. The probability density function for GNB $P(x|C_i)$ for a feature x given a class C_i, with μ the mean and σ the standard deviation of the feature's distribution, is given by the Gaussian distribution (Equation 2).

$$P(x|C_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{2}$$

A **DT** is a tree-like model where each node represents a decision based on a feature, leading to subsequent nodes. The tree building process (model training) recursively splits the dataset based on the most significant features, creating a tree structure that can be used for classification. The Gini Index is used as impurity measure, and is given by Equation 3, where P_i is the proportion of instances of class i in a node.

$$\text{Gini Index} = 1 - \sum_{i=1}^{C} P_i^2 \tag{3}$$

RF is an ensemble learning method that constructs a multitude of decision trees at training time. It combines the predictions of individual trees to improve overall accuracy and control overfitting. The entropy for a decision tree (Entropy(S)) in the forest, which plays a vital role in splitting the data, is calculated by using Equation 4, where P_i denotes the probability of class i.

$$\text{Entropy(S)} = -\sum_{i=1}^{C} P_i \cdot \log_2(P_i) \tag{4}$$

MLP is a type of artificial neural network with multiple layers of nodes. It is a powerful model capable of learning complex patterns. The output of a node in the network with activation function f is given by Equation 5.

$$\text{Output} = f\left(\sum_{i=1}^{n} w_i \cdot x_i + b\right) \tag{5}$$

where x_i is an input feature, w_i is the weight associated with input x_i and b is the bias term.

SVM is a powerful classification algorithm that aims to find a hyperplane that best separates data points of different classes. The equation for the hyperplane is provided in Equation 6.

$$\overrightarrow{\mathbf{w}} \cdot \overrightarrow{\mathbf{x}} + b = 0 \tag{6}$$

$\overrightarrow{\mathbf{w}}$ is the weight matrix, $\overrightarrow{\mathbf{x}}$ is the input feature, and b is the bias term.

kNN is a simple and intuitive classification algorithm that classifies data points based on the majority class among their *k* nearest neighbors. New data points or cases are classified based on similarity or distance measure (such as Euclidean distance, Manhattan distance, Minkowski distance, etc). No specific equation is involved in the training phase; it operates by finding the majority class among the *k*-nearest neighbors in the feature space.

LR is a linear model for binary classification that uses the logistic function to model the probability of a binary outcome. The logistic function is given by Equation 7.

$$P(Y = 1|X) = \frac{1}{1 + e^{-(b_0 + b_1 X_1 + b_2 X_2 + \ldots + b_n X_n)}} \tag{7}$$

$P(Y = 1|X)$ is the probability of class 1 given input features X, b_0 is bias term, $b_1, b_2, ..., b_n$ are the coefficients for input features $X_1, X_2, ..., X_n$ and e^x is the exponential function.

Five metrics are used to evaluate the performance of classifiers: (1) accuracy, (2) recall, (3) precision, (4) F1 score and (5) Area Under the ROC (Receiver Operating Characteristic) Curve (AUC). The five measures are defined as follows:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{8}$$

$$Precision(P) = \frac{TP}{TP + FP} \tag{9}$$

$$Recall(R) = \frac{TP}{TP + FN} \tag{10}$$

$$F1 = 2 \times \frac{P \times R}{P + R} \tag{11}$$

$$\text{AUC} = \int_0^1 \text{TPR}(d\text{FPR}) \tag{12}$$

whereas *TP*, *TN*, *FP* and *FN* stand for true positive, true negative, false positive and false negative respectively. In equation 5, *TPR* represents the recall (*R*) and *dFPR* is the derivative of the false positive rate (*FPR*), that is equal to: $\frac{FP}{FP+TN}$. All these measures are standard measures used for assessing how models perform in classification tasks.

5 Results

A computer with a 11th-generation Core i5 processor and 16 GB of RAM was utilized for carrying out experiments. Moreover, an open-source cross-platform Java package called SPMF [60], was used to analyze and discover patterns in the abstracted feature sequence datasets. SPMF offers implementations of more than 250 algorithms for various pattern discovery specializes tasks. For classification, Python is used, where a variety of libraries are utilized, including scikit-learn [61] for classifiers, NumPy for numerical computations, and Pandas for data manipulation. The performance of the classifiers is evaluated by using standard 10-fold cross validation. Next, the results obtained by applying the SPM algorithms on the datasets are discussed.

5.1 Frequent Patterns and Sequential Rules

Figure 2(a), (b) and (d) shows the frequent or top patterns in the whole datasets and in the respective classes (*Yes* or *No*) discovered by using Apriori algorithm. In the three datasets (IBM-1, HRD-2 and HRA-4), most of the features and their values in the whole dataset and in the two classes are almost the same. For IBM-1, the three differences are that (1) the *No* category has value 1 for the *Stock Option Level* feature, which is different from value 0 in the whole dataset and in the *Yes* category, (2) the *Yes* category has more *Married* employees, which is different from *Single* employees in the whole dataset and in the *No* category, (3) the *Yes* category contains value *Yes* for the feature *Over Time*, which is different from value *No* for *Over Time* in the whole dataset and in the *No* category. Similarly, in HRA-4, the difference is in the number of projects in the whole dataset and the two categories. For HRD-2, the top 10 frequent features in the whole dataset, and the *Yes* and *No* categories are the same.

For BCC-3, the count of all the features is shown in Figure 2(c). Apriori can also be used to discover frequent patterns of features and their values. However, frequent sets of feature values are unordered. Besides, Apriori does not ensure that feature values appear consecutively in a sequence. Thus, frequent patterns discovered by Apriori are uninteresting and do not provide any useful information. As Apriori ignores the sequential relationship among feature values, it cannot discover sequential patterns. The results for the TKS algorithm show that it overcomes this drawback of Apriori, as presented next.

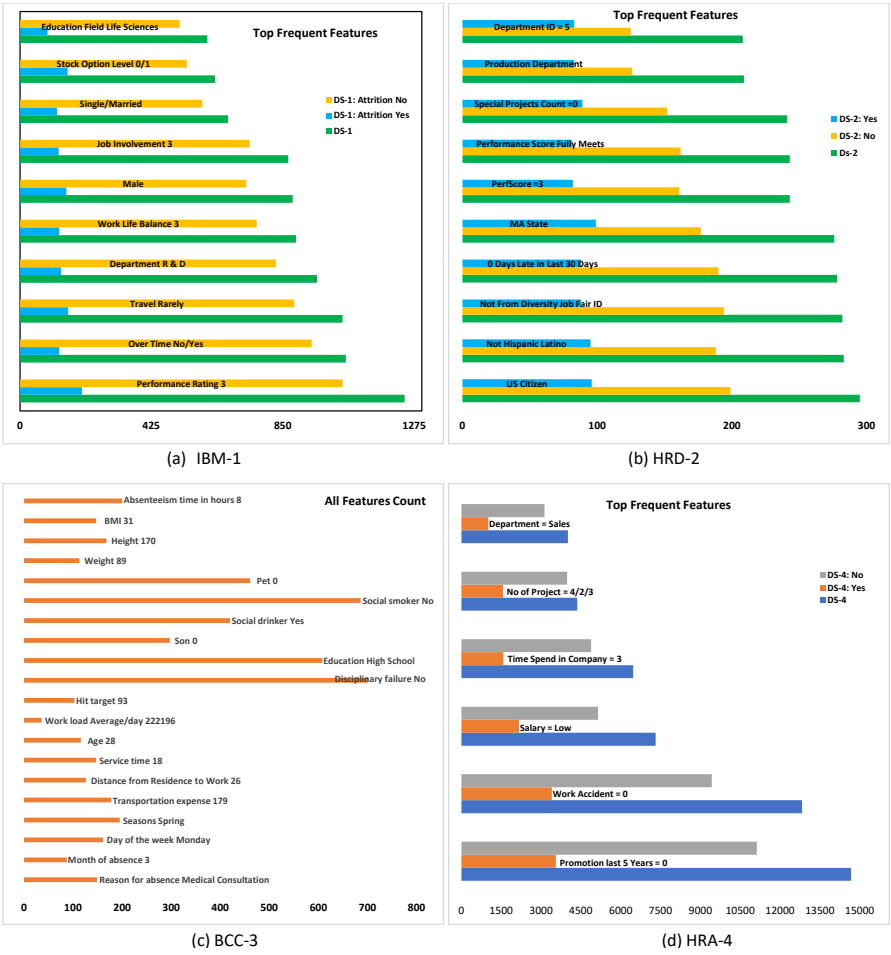


Fig. 2: Frequent features discovered in four datasets

Some frequent patterns of features discovered by the TKS algorithm in the four datasets, of varied lengths are shown in Table 6 and Table 7 respectively. These sequential patterns provide very useful information related to frequent occurrences of

Table 6: Frequent sequential patterns extracted by the TKS algorithm

IBM-1: Yes	**%**
Male, PerformanceRating: 3	54
PerformanceRating: 3, StockOptionLevel: 0	53.16
PerformanceRating: 3, WorkLifeBalance: 3	46
Single, StockOptionLevel: 0	50.63
OverTime: Yes, PerformanceRating: 3	43.88
TravelRarely, StockOptionLevel: 0	43.45
Male, PerformanceRating: 3, WorkLifeBalance: 3	27.84
JobInvolvement: 3, JobLevel: 1, PerformanceRating: 3	26.16
TravelRarely, Male, PerformanceRating: 3, Stock OptionLevel: 0	24.47
R & D, Male, JobLevel: 1, PerformanceRating: 3	22.78
IBM-1: No	**%**
Over Time: No, PerformanceRating: 3	64.63
TravelRarely, PerformanceRating: 3	61.23
Male, WorkLifeBalance: 3	36.33
Married, OverTime: No	36.17
TravelRarely, Male, OverTime: No	33.33
NumCompaniesWorked: 1, PerformanceRating: 3	28.46
TravelRarely, JobInvolvement: 3, OverTime: No, PerformanceRating: 3	27.65
R & D, EducationField: Medical	25.62
OverTime: No, PerformanceRating: 3, YearsSinceLastPromotion: 0	24.49
JobInvolvement: 3, OverTime: No, PerformanceRating: 3, WorkLifeBalance: 3	23.27
HRD-2: Yes	**%**
State: MA, Dept: Production	79.80
PerfScoreID: 3, PerformanceScore: Fully Meets, DaysLateLast30Days: 0,	76.92
HispanicLatino: No, DaysLateLast30Days: 0	75.96
State: MA, Citizen: US, Dept: Production, SpecialProjectsCount: 0	75
DeptID: 5, State: MA, Citizen: US, Dept: Production, SpecialProjectsCount: 0	75
DeptID: 5, State: MA, HispanicLatino: No, Dept: Production, SpecialProjectsCount: 0	73.07
EmpStatusID 5, Citizen: US, SpecialProjectsCount: 0	69.23
DeptID: 5, State: MA, Citizen: US, HispanicLatino: No, Dept: Production, SpecialProjectsCount: 0	68.26
EmpStatusID: 5, DeptID: 5, State: MA, Citizen: US, SpecialProjectsCount: 0	67.30
FromDiversityJobFairID: 0, State: MA, HispanicLatino: No, Dept: Production, SpecialProjectsCount: 0, DaysLateLast30Days: 0	55.76
HRD-2: No	**%**
Citizen: US, N/A-StillEmployed, LastPerformanceReviewDate: 2019	95.65
Citizen: US, N/A-StillEmployed, LastPerformanceReviewDate: 2019, DaysLateLast30Days: 0	87.92
FromDiversityJobFairID: 0, Citizen: US, HispanicLatino: No, N/A-StillEmployed, LastPerformanceReviewDate: 2019	82.12
PerfScoreID: 3, N/A-StillEmployed, PerformanceScore: FullyMeets, LastPerformanceReviewDate: 2019, DaysLateLast30Days: 0	76.81
PerfScoreID: 3, Citizen: US, PerformanceScore: FullyMeets, LastPerformanceReviewDate: 2019, DaysLateLast30Days: 0	73.42
N/A-StillEmployed, SpecialProjectsCount: 0, LastPerformanceReviewDate: 2019	72.94
EmpStatusID: 1, PerfScoreID: 3, FromDiversityJobFairID: 0, HispanicLatino: No, N/A-StillEmployed, PerformanceScore: FullyMeets, LastPerformanceReviewDate: 2019, DaysLateLast30Days: 0	57.97
DeptID: 5, Citizen: US, N/A-StillEmployed, Dept: Production, SpecialProjectsCount: 0, LastPerformanceReviewDate: 2019	57
EmpStatusID: 1, DeptID: 5, FromDiversityJobFairID: 0, State: MA, N/A-StillEmployed, Dept: Production, SpecialProjectsCount: 0, LastPerformanceReviewDate: 2019, DaysLateLast30Days: 0	45.89

feature values in the datasets. For example, in IBM-1 the pattern *Single, Stock Option Level: O* occurs for 50.63% of employees that left the company. Similarly, the sequential pattern *R & D, Male, Job Level: 1, Performance Rating: 3* occurs with a frequency of 22.78%. Similarly, for employees that did not leave, the sequences *Over Time: No, Performance Rating: 3* and *Job Involvement: 3, Over Time: No, Per-*

formance Rating: 3, Work Life Balance: 3 occur with frequencies of 64.63% and 23.27%, respectively.

Table 7: Frequent sequential patterns extracted by the TKS algorithm

BCC-3: Class A	**%**
ReasonForAbsence: No, AbsenteeismTimeInHours: 0	97.22
ReasonForAbsence: No, DisciplinaryFailure: Yes, AbsenteeismTimeInHours: 0	88.88
ReasonForAbsence: No, DisciplinaryFailure: Yes, SocialSmoker: No, AbsenteeismTimeInHours: 0	75
DisciplinaryFailure: Yes, Education: High School, SocialDrinker: Yes, SocialSmoker: No	55.55
Seasons: Spring, DisciplinaryFailure: Yes, Education: High School, SocialSmoker: No, AbsenteeismTimeInHours: 0	36.11
DayofWeek: Tuesday, Education: High School, SocialDrinker: Yes, AbsenteeismTimeInHours: 0	25
HitTarget: 88, DisciplinaryFailure: Yes, AbsenteeismTimeInHours: 0	13.88
BCC-3: Class B	**%**
DisciplinaryFailure: No, SocialSmoker: No	92.52
SocialSmoker: No, Pet: 0	57.16
Education: High School, SocialDrinker: Yes, SocialSmoker: No, Pet: No	37.53
TransportationExpenses: 179, DisciplinaryFailure: No, Son: 0, SocialSmoker: No	22.89
ServiceTime: 18, DisciplinaryFailure: No, SocialDrinker: Yes, SocialSmoker: No, Pet: 0, BodyMassIndex: 31	20.56
DistanceFromResidenceToWork: 51, ServiceTime: 18: Age: 38, Education: High School, Son: No, Social Drinker: Yes, SocialSmoker: No, Weight: 89	16.82
BCC-3: Class C	**%**
DisciplinaryFailure: No, Education: High School, SocialSmoker: No	79.03
SocialDrinker: Yes, SocialSmoker: No	66.12
Seasons: Winter, DisciplinaryFailure: No, Education: High School	33.87
DayOfTheWeek: Monday, DisciplinaryFailure: No, Education: High School, SocialDrinker: Yes, SocialSmoker: No	29.03
ServiceTime: 13, DisciplinaryFailure: No, Education: High School, SocialDrinker: Yes	14.54
TransportationExpense: 155, DistanceFromResidenceToWork: 12, Age: 34, DisciplinaryFailure: No, Pet: 0, Weight: 95, Height: 196	12.9

HRA-4: Yes	**%**
NumberProject: 2, TimeSpendCompany: 3, WorkAccident: 0, PromotionLast5Years: 0	40.49
NumberProject: 2, PromotionLast5Years: 0	43.60
NumberProject: 2, WorkAccident: 0, PromotionLast5Years: 0, Smoker: No	41.55
NumberProject: 2, TimeSpendCompany: 3, WorkAccident: 0, Salary: Low	25.09
NumberProject: 2, TimeSpendCompany: 3, WorkAccident: 0, PromotionLast5Years: 0, Salary: Low	24.97
TimeSpendCompany: 4, WorkAccident: 0, PromotionLast5Years: 0	24.05
TimeSpendCompany: 5, PromotionLast5Years: 0	25.28
NumberProject: 2, TimeSpendCompany: 3, WorkAccident: 0, PromotionLast5Years: 0, Department: Sales, Salary: Low	8.26
NumberProject: 2, TimeSpendCompany: 3, WorkAccident: 0, PromotionLast5Years: 0, Department: Sales, Salary: Medium	3.66
SatisfactionLevel: 0.37, NumberProject: 2, TimeSpendCompany: 3, WorkAccident: 0, PromotionLast5Years: 0, Salary: Low	1.26
HRA-4: No	**%**
WorkAccident: 0, PromotionLast5Years: 0	80.50
PromotionLast5Years: 0, Salary: Low	44.55
WorkAccident: 0, Salary: Medium	37.32
WorkAccident: 0, PromotionLast5Years: 0, Salary: Medium	36.14
NumberProject: 4, PromotionLast5Years: 0	33.62
TimeSpendCompany: 3, WorkAccident: 0, PromotionLast5Years: 0, salary: Low	16.08
NumberProject: 4, PromotionLast5Years: 0, Salary: Low	15.92
NumberProject: 4, WorkAccident: 0, PromotionLast5Years: 0, Salary: Low	13.07
NumberProject: 4, TimeSpendCompany: 3, WorkAccident: 0, PromotionLast5Years: 0	12.67
NumberProject: 3, TimeSpendCompany: 3, WorkAccident: 0, PromotionLast5Years: 0, Salary: Low	5.57

Table 8: Sequential rules extracted by the ERMiner algorithm

IBM-1			
Antecedents	**Consequents**	**Sup.**	**Conf.**
PerformanceRating: 3 (StockOptionLevel: 0)	Attrition: Yes	200 (154)	1 (1)
TravelRarely (JobLevel: 1)	Attrition: Yes	156 (146)	1 (1)
R & D (OverTime: Yes)	Attrition: Yes	133 (127)	1 (1)
PerformanceRating: 3, StockOptionLevel: 0	Attrition: Yes	126	1
Single, PerformanceRating: 3, StockOptionLevel: 0	Attrition: Yes	95	1
R & D, JobLevel: 1, PerformanceRating: 3	Attrition: Yes	80	1.0
Male	Attrition: Yes, PerformanceRating: 3	128	0.85
PerformanceRating: 3	Attrition: Yes, StockOptionLevel: 0	126	0.63
TravelRarely	Attrition: Yes, PerformanceRating: 3	134	0.85
TravelRarely, Male	Attrition: Yes, PerformanceRating: 3	89	0.87
PerformanceRating: 3 (OverTime: No)	Attrition: No	1,044 (944)	1 (1)
TravelRarely (R & D)	Attrition: No	887 (828)	1 (1)
WorkLifeBalance: 3 (JobInvolvement: 3)	Attrition: No	766 (743)	1 (1)
OverTime: No, PerformanceRating: 3	Attrition: No	797	1
TravelRarely, PerformanceRating: 3	Attrition: No	755	1
TravelRarely, OverTime: No, PerformanceRating: 3	Attrition: No	579	1
TravelRarely	Attrition: No, PerformanceRating: 3	755	0.85
R & D	Attrition: No, OverTime: No, PerformanceRating: 3	524	0.63
TravelRarely, OverTime: No	Attrition: No, PerformanceRating: 3	579	0.85
R & D, OverTime: No	Attrition: No, PerformanceRating: 3	524	0.83
HRD-2			
Antecedents	**Consequents**	**Sup.**	**Conf.**
State: MA (Citizen: US)	Terminated: Yes	99 (96)	1 (1)
HispanicLatino: No (SpecialProjectsCount: 0)	Terminated: Yes	95 (89)	1 (1)
DaysLateLast30Days: 0 (EmpStatusID: 5)	Terminated: Yes	88 (88)	1 (1)
State: MA	Terminated: Yes, Citizen: US	92	0.92
DeptID: 5	Terminated: Yes, State: MA, Department: Production	83	1
State: MA, Citizen: US	Terminated: Yes	92	1
State: MA, Citizen: US, HispanicLatino: No	Terminated: Yes	83	1
State: MA	Terminated: Yes, Department: Production, SpecialProjectsCount: 0	83	0.83
DeptID: 5, State: MA, Department: Production	Terminated: Yes, SpecialProjectsCount: 0	83	1
DeptID: 5, State: MA	Terminated: Yes, Department: Production	83	1
N/A-StillEmployed (HispanicLatino: No)	Terminated: No	207 (188)	1 (1)
LastPerformanceReviewDate: 2019 (DaysLateLast30Days: 0)	Terminated: No	206 (190)	1 (1)
Citizen: US (FromDiversityJobFairID: 0)	Terminated: No	199 (194)	1 (1)
FromDiversityJobFairID: 0, N/A-StillEmployed	Terminated: No	194	1
Citizen: US, N/A-StillEmployed	Terminated: No, LastPerformanceReviewDate: 2019	198	0.99
Citizen: US, N/A-StillEmployed, LastPerformanceReviewDate: 2019	Terminated: No	198	1
FromDiversityJobFairID: 0, N/A-StillEmployed, LastPerformanceReviewDate: 2019	Terminated: No	193	1
N/A-StillEmployed	Terminated: No, LastPerformanceReviewDate: 2019	206	0.99
FromDiversityJobFairID: 0	Terminated: No, N/A-StillEmployed, LastPerformanceReviewDate: 2019	193	0.99
Citizen: US	Terminated: No, N/A-StillEmployed, LastPerformanceReviewDate: 2019	198	0.99

Table 8 and 9 show the relationships among frequent features and their attributes that are identified in each dataset via the ERMiner algorithm, revealing intricate patterns that illuminate relationships between features and their values. It was observed that different datasets require different parameter settings (*minsup* and *minconf*) before they start giving sequential rules. For example, for IBM-1, the threshold for confidence (*minconf*) is set to 63%. A rule Y (antecedent) $\rightarrow Z$ (consequent) with a 63% confidence means that Y's feature values is followed by Z's feature values at least 63% of the times when a sequence contains the antecedent Y. The first three rules in each dataset represent the six most dominant feature values in the four datasets. The first rule in IBM-1 indicates that the feature *PerformanceRating: 3* or *Stock Option Level: 0* is followed by the feature *Attrition: Yes*. Similarly, the seventh rule indicates that *Male* is followed by *Attrition: Yes* and *PerformanceRating: 3* 128 (54%) times

Table 9: Sequential rules extracted by the ERMiner algorithm

BCC-3			
Antecedent(s)	**Consequent(s)**	**Sup.**	**Conf.**
ReasonForAbsence: 0 (Education: High School)	AbsenteeismTimeInHours: 0	35 (33)	1 (1)
SocialDrinker: 1 (Pet: 0)	AbsenteeismTimeInHours: 0	23 (21)	1 (1)
DiscipinaryFailure: Yes (Social Smoker: No)	AbsenteeismTimeInHours: 0	32 (31)	1 (1)
ReasonforAbsence: No, DisciplinaryFailure: Yes	AbsenteeismTimeInHours: 0	32	1
ReasonForAbsence: No, Education: High School, SocialSmoker: No	AbsenteeismTimeInHours:0	27	1
ReasonForAbsence: No, DisciplinaryFailure: Yes, Education: High School, SocialSmoker: No	AbsenteeismTimeInHours:0	24	1
ReasonForAbsence: No, Education: High School, SocialDrinker: Yes, SocialSmoker: No	AbsenteeismTimeInHours:0	22	1
ReasonForAbsence: No, DisciplinaryFailure: Yes, Education: High School, SocialDrinker: Yes, SocialSmoker: No, Pet:0	AbsenteeismTimeInHours: 0	15	1
DiscipinaryFailure: No (Social Smoker: No)	AbsenteeismTimeInHours: 8	200 (180)	0.31 (0.30)
DiscipinaryFailure: No (SocialSmoker: No)	AbsenteeismTimeInHours: 2	155 (117)	0.24 (0.22)
DiscipinaryFailure: No (Social Smoker: No)	AbsenteeismTimeInHours: 3	110 (105)	0.17 (0.17)
DisciplinaryFailure: No, Education: High School	AbsenteeismTimeInHours: 2	117	0.22
DisciplinaryFailure: No, SocialSmoker: No	AbsenteeismTimeInHours: 2	145	0.24
DisciplinaryFailure: No, SocialSmoker: No, Pet: 0	AbsenteeismTimeInHours: 3	67	0.18
DisciplinaryFailure: No, Son: 0, SocialSmoker: No, Pet: 0	AbsenteeismTimeInHours: 2	78	0.30
DisciplinaryFailure: No, Education: High School, SocialSmoker: No, Pet: 0	AbsenteeismTimeInHours: 2	68	0.23
DisciplinaryFailure: No, Education: High School, SocialDrinker: Yes, SocialSmoker: No, Pet: 0,	AbsenteeismTimeInHours: 8	66	0.27
DiscipinaryFailure: No (SocialSmoker: No)	AbsenteeismtimeInHours: 16	19 (17)	0.31 (0.29)
Education: High School (SocialDrinker: Yes)	AbsenteeismTimeInHours: 16 (24)	14 (14)	0.26 (0.31)
Pet: 0 (Son: 0)	AbsenteeismTimeInHours: 16	8 (14)	0.38 (0.31)
DisciplinaryFailure: Yes, SocialSmoker: No	AbsenteeismTimeInHours: 24	18	0.28
Eduction: High School, SocialDrinker: Yes	AbsenteeismTimeInHours:24	14	0.31
DisciplinaryFailure: No, Education: High School, SocialDrinker: Yes, SocialSmoker: No	AbsenteeismTimeInHours:24	14	0.34
DisciplinaryFailure: No, Education: High School, SocialDinker: Yes, SocialSmoker: No, Pet: 0	AbsenteeismTimeInHours: 16	8	0.27
DayoftheWeek: Wednesday, DisciplinaryFailure: No, Education: High School, SocialDrinker: Yes, SocialSmoker: No	AbsenteeismTimeInHours:24	8	0.66
Seasons: Winter, DisciplinaryFailure: No, Education: High School, SocialDrinker: Yes, SocialSmoker: No, Pet: 0	AbsenteeismTimeInHours: 24	9	0.65

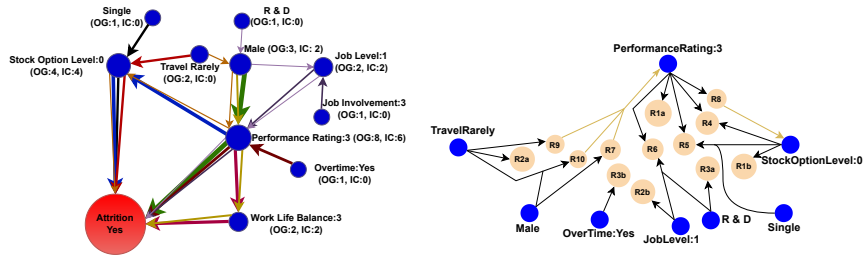
HRA-4			
Antecedents	**Consequents**	**Sup.**	**Conf.**
PromotionLast5years: 0 (WorkAccident: 0)	Left: Yes	3,552 (3,402)	1 (1)
TimeSpendCompany: 3 (NumberProjects: 2)	Left: Yes	1,583 (1,567)	1 (1)
Smoker: No (Salary: Low)	Left: Yes	3,567 (2,172)	1(1)
WorkAccident: 0	Left: Yes, Salary: Low	2,077	0.61
TimeSpendCompany: 3	Left: Yes, PromotionLast5Years: 0	1,569	0.99
NumberProjects: 2, TimeSpendCompany: 3	Left: Yes	1,527	1
NumberProjects: 2, WorkAccident: 0	Left: Yes, Smoker: No	581	0.99
WorkAccident: 0, Salary: Low,	Left: Yes	2,003	1
WorkAccident: 0, PromotionLast5Years: 0, Salary: Low	Left: Yes	2,066	1
NumberProjects: 5, TimeSpendCompany: 5, PromotionLast5Years: 0	Left: Yes	420	1
WorkaAccident: 0, PromotionLast5Years: 0	Left: Yes, Salary: Low	2,066	0.60
PromotionLast5Years: 0 (WorkAccident: 0)	Left: No	11,128 (9,428)	1 (1)
TimeSpendCompany: 3 (NumberProjects: 3)	Left: No	4,884 (3,983)	1(1)
Smoker: No (Salary: Low)	Left: No	11,380 (5,144)	1 (1)
WorkAccident: 0	Left: No, PromotionLast5Years: 0	9,200	0.97
TimeSpendCompany: 3	Left: No, PromotionLast5Years: 0	476	0.97
TimeSpendCompany: 3	WorkAccident: 0, Left: No, PromotionLast5Years: 0	3,967	0.81
NumberProject: 3, WorkAccident: 0	Left: No, Smoker: No	3,260	0.99
TimeSpendCompany: 4, WorkAccident: 0	Left: No	1,396	1
WorkAccident: 0, PromotionLast5Years: 0, Salary: Low	Left: No	4,162	1

with a confidence of 85%. In other words, the feature(s) from a rule's antecedent can be viewed as implying the features from the consequent.

The tenth rule found in IBM-1 indicates that the features *TravelRarely* and *Male* are followed by *Attrition Yes*, and *Performance rating 3* respectively. Similarly, the eighth rule in HRD-2 suggests that when an employee is in Massachusetts (*State: MA*) and marked as "*Terminated: Yes*," is associated with the "*Production*" department, and lacks special projects (*SpecialProjectsCount: 0*), a significant trend is observed. With a support of 83, this common pattern reflects its prevalence in the dataset. The confidence of 0.83 further highlights the strong correlation among these attributes, indicating a 83% likelihood of employee termination when these conditions are met. For BCC-3, *absenteeism time in hours: 0* is found for all rules in Class A.

To gain more insights, the patterns and rules are visualized in Figure 3. One frequent pattern *Travel Rarely, Male, Performance Rating: 3, Stock Option Level: 0* is represented by four dark orange arrows from the blue nodes (features) to the red node (Figure 3(a)).

For rules, take the example of rule number ten, represented by **R10** (Figure 3(b)). There are three blue nodes *TravelRarely*, *Male* and *PerformanceRating: 3*. These three nodes along with the yellow node for *Attrition: Yes* forms a rule where the antecedents are *TravelRarely* and *Male* and the consequents are *Attrition: Yes* and *PerformanceRating: 3*. The antecedent nodes have outgoing arrows toward the **R10** node. Similarly there is an outgoing arrow from **R10**, towards the consequent *PerformanceRating: 3* node. For this particular example, the most important feature with value is *Performance Rating of 3*, followed by *Stock Option Level of 0*, *Male*, *Job Level of 1*, *Work Life Balance of 3* respectively.



(a) Sequential patterns of features

(b) Sequential rules of features

Fig. 3: Graphical representation of frequent sequential patterns and rules discovered in a dataset (IBM-1, Attrition: *Yes*). Blue nodes in (a) and (b) represent features and their values and yellow nodes in (b) represent the class *Yes* for Attrition. Arrows size in (a) and yellow color nodes size in (b) is according to the support values. In (a) OG stands for number of outgoing arrows and IC stands for number of incoming arrows. Blue color nodes in (a) size is according to number of IG and OG. Arrows with the same color in (a) represent a pattern. The features with black arrow in (b) are antecedents and yellow nodes represent a consequent (Attrition: *Yes*). Yellow arrows represent the second consequent in the rule. R = rule

In summary, TKS and ERMiner provide some interesting relationships and dependencies between features and their values. The obtained results indicate that the effectiveness of SPM algorithms is directly correlated with the total number of features in a dataset. SPM algorithms allow one to find not only the frequent features and their relationships with each other but also the value of features that might play an important role in employee attrition and absenteeism.

5.2 Classification Results

This section discusses the experimental results for binary and MC classification on four datasets. The eight classifiers were used in two cases:

Case 1: The original datasets are used in this case without any preprocessing, meaning all the sequences that contain missing values for features are also considered. To overcome class imbalance in the datasets, the SMOTE algorithm [54] for oversampling is used.

Case 2: The frequent sequential patterns of feature values obtained with TKS are used in the classification process. After pattern discovery, the frequent patterns are pre-processed to ensure that each pattern contain at least 3 to 4 distinct frequent feature values.

The two cases are considered to compare the performance of classifiers when used on the original features and frequent sequential patterns of features.

For the classification in both cases, hyperparameters tuning was used to determine the best-fit parameters for the eight classifiers. The best hyperparameters for each classifier were found by varying them iteratively while examining the classifier's ACC result on the datasets. The parameters for which a classifier achieved the highest results were selected as its hyperparameters (Table 10).

Table 10: Hyperparamters of each classifier

Model	Parameters
MNB	$\alpha = 1$
GNB	default (no significant hyperparameters)
DT	criterion: gini, splitter: best, max depth: none, min samples split: 2, min samples leaf: 1
RF	estimators: 100, criterion: gini, max depth: none, min samples split: 2, min samples leaf: 1
MLP	hidden layer size: 600, activation: tanh, optimizer: Adam, $\alpha = 0.0001$, learning rate: invscaling, learning rate init: 0.001
SVM	C: 1, kernel: rbf, degree: 3, gamma: scale
kNN	n neighbors: 2, weight scheme: uniform, algorithm: auto, leaf size: 30, distance metric: euclidean
LR	C: 1, solver: lbfgs, max iterations: 100

Table 11 provides the classification results for case 1 (all features are used for classification). The format $\frac{Acc}{Time(Sec.)}$ is used for classifiers. For example the entry $\frac{51.9}{0.10}$ represents that the classifier MNB achieved an accuracy of 51.9% on IBM-1 and took 0.10 seconds to terminate. The results for HRD-2 is not included in the table because some classifiers such as SVM, RF achieved 100% accuracy. The main reasons for this is that HRD-2 contains very few sequences (311) that are insufficient

for classifiers training. Interestingly, sixe classifiers (MNB, GNB, MLP, SVM, kNN, LR) performed better without SMOTE. DT and RF performed better when SMOTE is applied on the datasets. The highest accuracy of 85.5%, achieved by RF, on the IBM-1 dataset without SMOTE is better than the highest accuracy achieved in studies [16, 39, 45]. Whereas, the highest accuracy of 90.5%, achieved by RF, on the IBM-1 dataset with SMOTE is better than the highest accuracy achieved in studies [1, 2, 14, 15, 22, 40, 43, 44, 48–50]. For BCC-3, the highest accuracy of 95.7%, achieved by RF with SMOTE is better than the highest accuracy achieved in studies [6, 7, 46, 47].

Table 11: Accuracy of the classifiers (Case 1). Binary classification in IBM-1 and HRA-4, MC classification in BCC-3

	without SMOTE							
Dataset	MNB	GNB	DT	RF	MLP	SVM	kNN	LR
IBM-1	51.9	78.8	74.3	**85.5**	79.9	83.8	81.2	84.4
	0.10	0.09	0.55	11.68	3.37	1.09	0.28	5.87
BCC-3	72.4	86.3	82.7	**88.1**	86.6	86.8	86.6	86.8
	0.06	0.07	0.09	3.30	0.58	0.50	0.36	0.85
HRA-4	76.1	68.7	97.7	**98.9**	95.3	94.8	94.1	79.1
	0.17	0.18	0.91	15.11	34.67	188.61	0.83	0.33
Ave.	66.8	77.9	84.9	**90.8**	87.2	88.4	87.3	83.4
	0.11	0.11	0.51	10.3	12.87	63.4	0.49	2.35
	with SMOTE							
Dataset	MNB	GNB	DT	RF	MLP	SVM	kNN	LR
IBM-1	55.1	71.8	75.9	**90.5**	55.8	59.1	76.1	71.7
	0.12	0.15	0.91	14.27	8.59	5.74	0.52	6.20
BCC-3	74.0	82.1	94.1	**95.7**	33.7	46.3	77.9	75.2
	0.10	0.07	0.63	11.32	1.67	13.21	0.29	1.32
HRA-4	67.8	64.6	98.0	**98.9**	95.7	94.6	95.9	79.5
	0.24	0.32	2.08	41.26	102.53	781.56	2.07	0.60
Ave.	65.6	72.8	89.3	**95.0**	61.7	66.6	83.8	75.4
	0.15	0.18	1.20	22.28	37.59	266.83	0.96	2.70

RF achieved the highest average accuracy of 95% on all datasets with SMOTE, and 90.8% on all datasets without SMOTE. The ranking of classifiers on datasets with SMOTE on the basis of average accuracy is in the order RF > DT > kNN > LR > GNB > SVM > MNB > MLP. In terms of computational time, GNB and MNB performed best. The ranking of classifiers on the basis of time is MNB > GNB > kNN > DT > LR > RF > MLP > SVM. RF performed better than DT, on overall, but RF was slow compared to DT. The detailed results for RF are listed in Table 12.

In Case 2, the classifiers exhibited notable improvement over Case 1 (see Table 13) by achieving excellent accuracies in all the three datasets. This underscores their proficiency in achieving remarkable accuracies on datasets with frequent patterns. All classifiers achieved their highest accuracy with IBM-1 patterns, followed by BCC-3 as compared to HRA-4. On average, the algorithms performed exceptionally well, with an average accuracy ranging from 95.3% to 99.9% and average runtimes from 0.481 to 127.30 seconds. The ranking of classifiers on the basis of accuracy on patterns discovered by TKS is in the order DT > LR > GNB > RF > SVM > MNB >

Table 12: Classification results for RF (Case 1)

	without SMOTE				
Dataset	ACC	P	R	F1	AUC
IBM-1	85.5	69.44	15.16	24.38	0.798
BCC-3	88.1	64.6	66.9	64.8	0.864
HRA-4	98.9	99.0	98.0	98.5	0.996
	with SMOTE				
Dataset	ACC	P	R	F1	AUC
IBM-1	90.5	91.5	89.4	89.6	0.968
BCC-3	95.7	95.7	95.7	95.7	0.993
HRA-4	98.9	98.7	98.7	98.7	0.997

Table 13: Classifiers accuracy for sequential patterns (Case 2). Binary classification in IBM-1 and HRA-4, MC classification in BCC-3

Dataset	MNB	GNB	DT	RF	MLP	SVM	kNN	LR
IBM-1	100	**100**	100	100	99.959	100	100	100
	0.121	0.078	0.125	3.085	2.855	4.378	0.428	0.251
BCC-3	99.913	100	**100**	100	99.980	100	99.713	100
	1.237	1.712	1.330	27.427	34.578	373.868	8.784	3.946
HRA-4	90.4	96.7	**99.7**	95.2	86.0	91.2	87.4	97.1
	0.085	0.087	0.206	3.591	1.574	3.660	0.407	0.979
Ave.	96.7	98.9	**99.9**	98.4	95.3	97.1	95.7	99.0
	0.481	0.625	0.553	11.367	13.01	127.30	3.20	1.725

Table 14: Classification results for DT (Case 2)

Dataset	ACC	P	R	F1	AUC
IBM-1	100	100	100	100	1.0
BCC-3	100	100	100	100	1.0
HRA-4	99.7	99.4	99.4	99.5	0.99

kNN > MLP. MLP performed lowest in both case 1 and case 2. Table 14 lists the overall results for DT, which performed best on patterns discovered by using TKS. The results presented in Tables 13 and 14 clearly demonstrate the advantage of using frequent patterns instead of using all the features in the classification process for employee attrition.

Lessons learned from the research conducted and obtained results are: (1) discovered sequential patterns and rules in employee data not only provide information for features that play an important role in employee attrition and absenteeism, but also about their values. (2) frequent sequential patterns of features and their values can be used efficiently for classification and detection in place of providing all the features. Table 3 lists the total number of features considered in each dataset: IBM-1 has 32 features, BCC-3 has 20 features and HRA-4 has 11 features. However, the sequential patterns discovered by the TKS algorithm contain 9 features for IBM-1 (71.8% reduction), 13 features for BCC-3 (35% reduction) and 9 features for HRA-4 (18% reduction).

5.2.1 Comparison

E(3A)CSPM is compared in this section with state-of-the-art approaches (published in 2020-2023) for employee attrition and absenteeism classification and detection (Table 15).

Table 15: E(3A)CSPM comparison with state-of-the-art approaches

Ref.	**Dataset(s) used**	Best Learning Model	ACC	P	R	F1	AUC
[1]	IBM-1	DT+LR	88.43	74	46	57	0.859
[2]	IBM-1	XGBoost	87.1	65	37.7	47.7	–
[3]	IBM-1	ESM1	97.6	99.8	95.1	97.5	0.975
[13]	Selfmade	RF	86	–	–	85.99	–
[14]	IBM-1	LR	87	78.5	66.5	70	–
[15]	IBM-1	LR	90.47	–	–	–	–
[16]	IBM-1	kNN	84	47	12	18.7	0.79
[17]	IBM-1, HRA-4, **Selfmade**	Voting Classifier (VC)	99	–	–	91	–
[18]	IBM-1	ANN+SVMSMOTE +BiasInitializer	96	96	96	96	–
[19]	IBM-1, **HRA-4**, Selfmade	Deep RF	98.7	99	95.5	97.2	0.99
[20]	IBM-1, **HRA-4**	stacker-top5-RF	99.3	99.1	98	98.5	0.99
[21]	SAS	NN	85	–	99	59	–
[22]	IBM-1	LR	87.96	–	–	31.26	85.01
[23]	Selfmade	EL1 and EL2	94.7	94.7	94.7	94.7	–
[36]	IBM-1	Gradient Boosting	95.05	71	71.5	77	–
[37]	IBM-1	XGBoost	–	–	–	–	0.86
[38]	IBM-1	Extra Trees	93	93	93	93	–
[39]	IBM-1	Gaussian NB	73.34	32.6	70.76	44.63	–
[40]	IBM-1	LR	86.73	63.89	46.94	54.11	0.84
[41]	HRA-4	KPCA+AdpK-means based	96.9	89.1	56.1	–	0.86
[42]	IBM-1	LightGBM	–	–	–	–	0.83
[43]	IBM-1	LR	88.09	88.5	66	48	–
[44]	IBM-1	Linear SVC	87.9	66.5	24.7	35.8	–
[45]	IBM-1	LR	81	43	82	56	–
[48]	IBM-1	LR	87.78	–	–	–	–
[49]	IBM-1	XGBoost	87.07	79.8	64.7	68.4	–
[50]	IBM-1	AdaBoost	87	78	66.5	70	0.80
[52]	ORACLE ERP	RF+PCA	99	88	99	93	–
[53]	HRA-4	DT+CS	98.6	97.7	96.4	–	–
[5]	BCC-3	Deep NN	97.5	97	97	97	–
[6]	BCC-3	KNN-Chy	92.3	87	68	75	–
[7]	BCC-3	MLP	83	–	72.5	–	–
[46]	BCC-3	Bagging+CFS	92	90	92	–	–
[47]	BCC-3	Multinomial LR	88	63	63.6	63	–
E(3A)CSPM	IBM-1, BCC-3, HRA-4	DT+TKS	**99.9**	**99.73**	**99.8**	**99.83**	**0.996**

Majority of prior studies used a single dataset. The maximum number of three datasets were used in [17, 19]. The bold datasets in column 2 of Table 15 are those datasets for which the corresponding learning model achieved the best results. For example, the study [17] achieved the highest accuracy of 99% on the selfmade dataset using voting classifier. For binary classification, the study [20] achieved the highest accuracy of 99.3%, by using stacker-top5-RF: an ensemble model obtained by stacking the top five base models with RF as the meta-estimator, followed by [17] with an accuracy of 99% and [19] with an accuracy of 98.7%. Note that we added average DT results from this work on three datasets for comparison as it performed better than other classifiers in case 2. ESM1 in [3] is the ensemble method where two classifiers (RF and ANN) are used as base models and LR as meta-learner). EL1 and EL2 ensemble learning models of [23] are made of 7 classifiers (GB, RF, NN,

kNN, SVM, NB, LR) and 3 classifiers (GB, RF, kNN) respectively. The results for the study [8] are not provided in the table as it evaluated the performance of classifiers for a different metric (the RMSE).

The study [53] found that DT classifier performed better when Chi-Square (CS) was used as compared to the Fisher score, Spearman correlation and R coefficient correlation. Similarly, the bagging algorithm performed better in [46] with correlation-based feature selection (CFS) compared to relief-based and information-gain-based feature selection algorithms. E(3A)CSPM(DT) results for both binary and MC classification and their average show that it outperforms other classifiers.

6 Conclusion

A novel SPM-based methodology (called E(3A)CSPM) is presented to analyze and classify employee attrition and absenteeism. Four diverse public datasets were used to investigate the effectiveness and generalization ability of E(3A)CSPM. The datasets were first abstracted, and SPM algorithms were then used to find frequent feature and their values, their frequent sequential patterns, and their frequent sequential rules. Discovered frequent patterns of feature values were then used for classification. Eight classifiers were used, and their performance was evaluated using five metrics. Obtained results suggest that DT performed better than others for binary and MC classification. From the obtained results, it was observed that limited (or short) sequences, containing only frequent patterns of feature and their values, can be used for reliable prediction and classification rather than using all the features. Moreover, E(3A)CSPM outperformed state-of-the-art approaches for employee attrition and absenteeism classification and detection.

This study has a number of limitations: (1) Retrospective and static datasets for employee attrition and absenteeism are analyzed with no guarantee for the standardization of features. The majority of the features in each dataset are different from each other and have no specific range for values. (2) There is no explanation for how the data was gathered and made public due to the online nature of the datasets; thus, we cannot completely rule out information collection bias in the study. (3) Patterns and rules discovered by algorithms require validation and verification from the industries and their HR.

In the future, other types of patterns-based classifiers could be developed. In particular, rules could be used to build an associative classifier. A comparison could then be done with the approach proposed in this paper based on frequent sequential patterns to compare the classification performance. Another interesting area of research for future work is to use emerging or contrast pattern mining [62] on the datasets to find contrasting frequent patterns of feature and their values and use these patterns for analysis and classification. Lastly, another research direction is to use SPM algorithms on dynamic datasets for employee attrition and absenteeism.

Conflict of Interest: Authors declare no conflict on interest.
Funding: Authors did not receive funding for this work.

References

1. A. Qutub, A. Al-Mehmadi, M. Al-Hssan, R. Aljohani and H.S. Alghamdi. Prediction of Employee Attrition Using Machine Learning and Ensemble Methods. *International Journal of Machine Learning and Computing*, 11(2), 110-114, 2021.
2. S. Gim and E.T. Im. A Study on Predicting Employee Attrition Using Machine Learning. In *Proceedings of BCD 2022*, vol 1075, pp. 55-69, Springer, 2023.
3. D. Chung, J. Yun, J. Lee and Y. Jeon. Predictive Model of Employee Attrition based on Stacking Ensemble Learning. *Expert Systems with Applications*, 215: 119364, 2023.
4. N. Lawrance, G. Petrides and M-A Guerry. Predicting Employee Absenteeism for Cost Effective Interventions. *Decision Support Systems*, 147: 113539, 2021.
5. S. A. A. Shah, I. Uddin, F. Aziz, S. Ahmad, M. A. Al-Khasawneh and M. Sharaf. An Enhanced Deep Neural Network for Predicting Workplace Absenteeism, *Complexity*, 5843932, 2020.
6. M. Skorikov, M. A. Hussain, M. R. Khan, M. K. Akbar, S. Momen, N. Mohammed and T. Nashin. Prediction of Absenteeism at Work using Data Mining Techniques. In *Proceedings of ICITR*, pp. 1-6, 2020.
7. J. M. G. Junior and F. M. Lopes. Interpretability with Relevance Aggregation in Neural Networks for Absenteeism Prediction. In *Proceedings of BHI*, pp. 01-04, 2022.
8. B. Hu. The Application of Machine Learning in Predicting Absenteeism at Work. In *Proceedings of CDS*, pp. 270-276, 2021.
9. U.S. Bureau of Labor Statistics. Job Openings and Labor Turnover - September 2023, available at: bls.gov/news.release/pdf/jolts.pdf, accessed on November 17, 2023.
10. K. Navarra. The real costs of recruitment. Society for Human Resource Management, available at: shrm.org/resourcesandtools/hr-topics/talent-acquisition/pages/the-real-costs-of-recruitment.aspx, accessed on November 19, 2023.
11. U.S. Bureau of Labor Statistics. Labor force statistics from the current population survey, available at: https://www.bls.gov/cps/cpsaat47.htm, accessed on November 19, 2023.
12. R. Punnoose and P. Ajit, Prediction of Employee Turnover in Organizations using Machine Learning Algorithms. *International Journal of Advanced Research in Artificial Intelligence*, 5(9): C5, 2016.
13. R. Joseph, S. Udupa, S. Jangale, K. Kotkar and P. Pawar. Employee Attrition Using Machine Learning And Depression Analysis. In *Proceedings of ICICCS*, pp. 1000-1005, 2021.
14. M. Maharana, R. Rani, A. Dev and A. Sharma, Automated Early Prediction of Employee Attrition in Industry Using Machine Learning Algorithms. In *Proceedings of ICRITO*, 2022, pp. 1-6, 2022.
15. G. Raja Rajeswari, R. Murugesan, R. Aruna, B. Jayakrishnan and K. Nilavathy. Predicting Employee Attrition through Machine Learning. In *Proceedings of ICOSEC*, pp. 1370-1379, 2022.
16. M. Atef, D. S. Elzanfaly and S. Ouf. Early Prediction of Employee Turnover Using Machine Learning Algorithms. *International Journal of Electrical and Computer Engineering Systems*, 13(2): 135-144, 2022.
17. N. B. Yahia, J. Hlel and R. C.-Palacios. From Big Data to Deep Data to Support People Analytics for Employee Attrition Prediction. *IEEE Access*, 9: 60447-60458, 2021.
18. S. Soner, A. A. Hussain, R. Khatri, S. K. Kushwaha, S. Mathariya and S. Bhayal. Predictive Deep Learning approach of employee attrition for imbalance datasets using SVMSMOTE algorithm with Bias Initializer, *PREPRINT*, 2022,
19. K. Gurler, B. K. Pak, V. C. Gungor. Deep Learning Based Employee Attrition Prediction. In *Proceedings of AIAI*, vol 675, 2023.
20. X. Wang and J. Zhi. A machine learning-based analytical framework for employee turnover prediction. *Journal of Management Analytics*, 8:3, 351-370, 2021.
21. F. K. Alsheref, I. E. Fattoh and W. M. Ead. Automated Prediction of Employee Attrition Using Ensemble Model Based on Machine Learning Algorithms. *Computational Intelligence and Neuroscience*, 7728668, 2022.
22. F. Guerranti and G. M. Dimitri. A Comparison of Machine Learning Approaches for Predicting Employee Attrition. *Applied Sciences*, 13(1): 267, 2023.
23. A. K. Biswas, R. Seethalakshmi, P. Mariappan and D. Bhattacharjee. An Ensemble Learning Model for Predicting the Intention to Quit among Employees using Classification Algorithms. *Decision Analytics Journal*, 9: 100335, 2023.
24. C. Aggarwal, M. Bhuiyan and M. Hasan. *Frequent Pattern Mining Algorithms: A Survey*. Springer, 2014.

25. P. Fournier-Viger, J. C. W. Lin, R. U. Kiran, Y. S. Koh and R. Thomas. A Survey of Sequential Pattern Mining. *Data Science and Pattern Recognition*, 1(1):54-77, 2017.
26. A. I. A. Aldine, M. Harzallah, G. Berio, N. Bèchet, A. Faour. Mining Sequential Patterns for Hypernym Relation Extraction. In *Proceedings of the TextMine'19*, pp. 21-24, 2019.
27. A. I. A. Aldine, M. Harzallah, G. Berio, N. BÃ©chet, A. Faour. A 3-phase Approach based on sequential Mining and Dependency Parsing for Enhancing Hypernym Patterns Performance. *The Knowledge Engineering Review*, 36: E13, 2021.
28. M. S. Nawaz, P. Fournier-Viger, A. Shojaee, H. Fujita. Using Artificial Intelligence Techniques for COVID-19 Genome Analysis. *Applied Intelligence*, 51(5): 3086-3103, 2021.
29. M. S. Nawaz, P. Fournier-Viger, M. Aslam, W. Li, Y. He and X. Niu. Using Alignment-free and Pattern Mining methods for SARS-CoV-2 Genome Analysis. *Applied Intelligence*, 53: 21920-21943, 2023.
30. M. Cheng, X. Jin, Y. Wang, X. Wang and J. Chen. A Sequential Pattern Mining Approach to Tourist Movement: The Case of a Mega Event. *Journal of Travel Research*, 62(6): 1237-1256, 2023.
31. L. Ni, W. Luo, N. Lu, W. Zhu. Mining the Local Dependency Itemset in a Products Network. *ACM Transactions on Management Information Systems*, 11(1):3:1-3:31, 2020.
32. M. S. Nawaz , P. Fournier-Viger, M. Z. Nawaz, G. Chen and Y. Wu. MalSPM: Metamorphic malware behavior analysis and classification using sequential pattern mining. *Computers & Security*, 118: 102741, 2022.
33. M Amiri, L. Mohammad-Khanli and R. Mirandola. A Sequential Pattern Mining Model for Application Workload Prediction in Cloud Environment. *Journal of Network and Computer Applications*, 105: 21-62, 2018,
34. H. Estiri, S. Vasey and S. N. Murphy. Transitive Sequential Pattern Mining for Discrete Clinical Data. In Proceedings of *AIME*, pp. 414-424, 2020.
35. M. S. Nawaz, P. Fournier-Viger, Y. He and Q. Zhang. PSAC-PDB: Analysis and Classification of Protein Structures. *Computers in Biology and Medicine*, 158: 106814, 2023.
36. V. Mehta and S. Modi. Employee Attrition System Using Tree Based Ensemble Method. In Proceedings of C2I4, pp. 1-4, 2021.
37. N. Darapaneni, R. N. Turaga, V. C. Shah, A. R. Paduri, D. Kumar R, M. Suram and V. Venkatraman. A Detailed Analysis of AI Models for Predicting Employee Attrition Risk. In *Proceedings of (R10-HTC)*, pp. 243-246, 2022.
38. A. Raza, K. Munir, M. Almutairi, F. Younas, N.M.S. Fareed. Predicting Employee Attrition Using Machine Learning Approaches. *Applied Sciences*, 12: 6424, 2022.
39. A. Habous, E. H. Nfaoui and Y. Oubenaalla. Predicting Employee Attrition using Supervised Learning Classification Models. In *Proceedings of ICDS*, pp. 1-5, 2021.
40. S. Y. Bansal, B. Kaur and J. R. Saini. A Novel Optimized Approach for Machine Learning Techniques for Predicting Employee Attrition. In *Proceedings of SMART GENCON*, pp. 1-9, 2022.
41. G. Pratibha and N. P. Hegde. HR Analytics: Early Prediction of Employee Attrition using KPCA and Adaptive K-means based Regression. In *Proceedings of ICPS*, pp. 11-16, 2022.
42. K. Sekaran and S. Shanmugam. Interpreting the Factors of Employee Attrition using Explainable AI. In *Proceedings of DASA*, pp. 932-936, 2022.
43. S. Gupta, G. Bhardwaj, M. Arora, R. Rani, P. Bansal and R. Kumar. Employee Attrition Prediction in Industries using Machine Learning Algorithms. In *Proceedings of INDIACom*, pp. 945-950, 2023.
44. F. Fallucchi, M. Coladangelo, R. Giuliano and E. William De Luca. Predicting Employee Attrition Using Machine Learning Techniques. *Computers*, 9: 86, 2020.
45. S. Najafi-Zangeneh, N. Shams-Gharneh, A. Arjomandi-Nezhad and S. H. Zolfani. An Improved Machine Learning-Based Employees Attrition Prediction Framework with Emphasis on Feature Selection. *Mathematics*, 9: 1226, 2021.
46. Amal Al-Rasheed. Identification of Important Features and Data Mining Classification Techniques in Predicting Employee Absenteeism at Work. *International Journal of Electrical and Computer Engineering*, 11(5): 4587-4596, 2021.
47. D. Naganaidu, Z. M. Khalid and S. Govindan. Prediction of Absenteeism at Work with Multinomial Logistic Regression Model. *Advances and Applications in Mathematical Sciences*, pp. 1479-1489, 2022.
48. N. Khalifa, M. Alnasheet and H. Kadhem, Evaluating Machine Learning Algorithms to Detect Employees' Attrition. In *Proceedings of AIRC*, pp. 93-97, 2-22, 2022.
49. K. M. Mitravinda and S. Shetty. Employee Attrition: Prediction, Analysis Of Contributory Factors And Recommendations For Employee Retention. In *Proceedings of ICWITE*, pp. 1-6, 2022.
50. P. J. Padmaja, D. Vinoodhini and K. Uma. Effective Classification Of Ibm Hr Analytics Employee Attrition Using Sampling Techniques. In *Proceedings of ICAECT*, pp. 1-6, 2022.

51. N. Silpa, V. V. R. Maheswara Rao, M. V. Subbarao, R. R. Kurada, S. S. Reddy and P. J. Uppalapati. An Enriched Employee Retention Analysis System with a Combination Strategy of Feature Selection and Machine Learning Techniques. In *Proceedings of ICICCS*, pp. 142-149, 2023.
52. A. B. W. Ali. Prediction of Employee Turn Over Using Random Forest Classifier with Intensive Optimized PCA Algorithm. *Wireless Personal Communications*, 119:3365-3382, 2021.
53. K. Naz. I. F. Siddiqui, J. Koo, M. A. Khan and N. M. F. Qureshi. Predictive Modeling of Employee Churn Analysis for IoT-Enabled Software Industry. *Applied Sciences*, 12: 10495, 2022.
54. N. V. Chawla, K. W. Bowyer, L. O. Hall and W. P. Kegelmeyer. SMOTE: Synthetic Minority Over-sampling Technique. *Journal Of Artificial Intelligence Research*, 16: 321-357, 2002.
55. I. R. White, P. Royston and A. M. Wood. Multiple Imputation using Chained Equations: Issues and Guidance for Practice. *Statistics in Medicine*, 30(4): 377-399, 2011.
56. P. Fournier-Viger, A. Gomariz, T. Gueniche, E. Mwamikazi and R. Thomas. TKS: Efficient mining of top-k sequential patterns. In: *Proceedings of ADMA*, pp. 109-120, 2013.
57. P. Fournier-Viger, T. Gueniche, S. Zida, and V. S. Tseng. ERMiner: Sequential rule mining using equivalence classes. In *Proceedings of IDA*, pp. 108-119, 2014.
58. C. C. Aggarwal. *Data Classification Algorithms and Applications*. 1st Edition, CRC Press, 2015
59. R. J. Urbanowicz and W. N. Browne. *Introduction to Learning Classifier Systems*. 1st Edition, Springer, 2017.
60. P. Fournier-Viger, J. C.-W. Lin, A. Gomariz, T. Gueniche, A. Soltani, Z. Deng H. T. Lam. The SPMF Open-Source Data Mining Library Version 2. In *Proceedings of ECML/PKDD*, pp. 36–40, 2016.
61. O. Kramer. Scikit-Learn. In: *Machine Learning for Evolution Strategies*. Studies in Big Data, vol 20. Springer, 2016.
62. S. Ventura and J. M. Luna, *Supervised Descriptive Pattern Mining*. Springer, 2018.