Array 30 (2026) 100762

Contents lists available at ScienceDirect

Array

journal homepage: www.elsevier.com/locate/array

A simple heuristic for the identification of the case ID attribute in unlabelled process mining event logs

André Vicente [a], Carlos Grilo [a,b], Rui Rijo [a,c], Ricardo Martinho [a,c,*]

[a] ESTG, Polytechnic University of Leiria, Morro do Lena, Alto do Vieiro, Leiria, 2411-901, Portugal
[b] CIIC - Computer Science and Communication Research Centre, Morro do Lena, Alto do Vieiro, Leiria, 2411-901, Portugal
[c] INESCC-DL - Institute for Systems Engineering and Computers at Coimbra – Delegation of Leiria, Morro do Lena, Alto do Vieiro, Leiria, 2411-901, Portugal

ARTICLE INFO

Keywords:
Process mining
Unlabelled event logs
Case ID identification
Heuristics

ABSTRACT

This study addresses the critical challenge of identifying and labelling the case ID attribute in unlabelled event logs, a fundamental task in process mining. Case IDs uniquely associate events with individual process instances, enabling accurate analysis and discovery of operational insights. Manual identification of case IDs is error-prone and labour-intensive, often hindering the scalability and reliability of process mining analyses. This paper introduces a novel heuristic method that automates case ID identification, improving efficiency and accuracy for diverse real-world datasets. The proposed heuristic leverages unique temporal patterns observed in event logs to distinguish case ID attributes from other attributes. It calculates a weighted average of temporal spans and applies customisable parameters to prioritise relevant attributes. The method was validated using 27 datasets from the Business Process Intelligence (BPI) Challenge, representing a variety of industries and event log complexities. Performance metrics, including success rates and computational efficiency, were benchmarked against existing approaches. The heuristic achieved an 85.2% top-1 success rate, and remains effective provided at least one repeating categorical attribute is present - a condition met by virtually all publicly available business and industrial logs. It consistently ranked case IDs among the top attributes even in challenging scenarios, such as cyclic processes and multi-correlated data. The method demonstrated robustness across diverse datasets, processing large event logs within seconds, highlighting its practicality for real-world applications. This research contributes an innovative and explainable approach to case ID identification that requires only raw event logs, contrasting with existing methods reliant on pre-labelled data or complex pipelines. Its simplicity, efficiency, and adaptability to various process types make it a valuable tool for advancing process mining capabilities.

1. Introduction

In the field of Process Mining (PM), event logs play a crucial role in discovering, understanding and improving business processes and the quality of their analyses. These event logs capture a wealth of information about the sequence of activities performed in a process, its assigned roles and participants and the quality of its data, providing insights into process execution, bottlenecks and deviations, conformance degree and potential improvements. Event logs consist of structured records of events or activities in business processes, documenting actions chronologically with timestamps and other relevant data about cost, human labour, and other resource expenditures. Nevertheless, the journey from raw data to event logs suitable for PM can be long, and addressed by a variety of methods and techniques [1]. This includes techniques for identification and extraction of the required event data, where the case ID attribute represents an essential element which uniquely identifies the events related to a certain process case.

However, in many real-world scenarios, these event logs often lack a dedicated labelled header line, including a case ID attribute identified. Usually, process engineers load a dataset into a PM tool, and manually identify, one by one, each column of this log, including not only its case ID, but also timestamps, activities, resources and costs, among other valuable data for process analysis. This identification heavily relies on the process engineer's expertise and knowledge of the process being evaluated. Identifying the case ID attribute is a critical pre-processing step that enables a correct and subsequent analysis, as well as the PM techniques to be applied effectively, namely process discovery, conformance and enhancement techniques. It can even be applied in automated identification and discovery of process case information in non-standard (event log) formats such as relational databases [2].

* Corresponding author at: ESTG, Polytechnic University of Leiria, Morro do Lena, Alto do Vieiro, Leiria, 2411-901, Portugal.
E-mail address: ricardo.martinho@ipleiria.pt (R. Martinho).

https://doi.org/10.1016/j.array.2026.100762
Received 17 March 2025; Received in revised form 25 October 2025; Accepted 16 March 2026
Available online 17 March 2026

In fact, PM tools such as ProM and Disco, among others, require, as a first step to process mining analyses, the loading of an event log dataset and the proper identification of a minimum set of attributes including case ID, activity and timestamp(s). This is usually done manually by process engineers and, with large attribute count event logs and inexplicit labelling, requires significant effort and time. Additionally, this manual identification and labelling of attributes, especially the case ID, is a daunting and error-prone task, since event logs often contain dozens or even hundreds of attributes, many of which have overlapping or ambiguous roles. Without clear labelling, process engineers must rely on domain knowledge and intuition to discern which attribute represents the case ID — a process that is inherently subjective and inconsistent [3].

Indeed, manual labelling does not scale. A multi-company survey [4] found that attribute labelling – including case-ID choice – consumed a median 28% of preparation effort. Other studies reported that data preparation accounts for 40 to 60% of the end-to-end PM effort, with case-ID identification recognised as the single most time-consuming step [5,6]. For instance, logs exported from operational databases or data lakes often arrive as flat Comma Separated Values (CSV) files with cryptic or missing headers. When dozens – or in some industrial historians, hundreds – of attributes are present, analysts must iteratively load the file into a PM tool, select a candidate column, inspect partial traces and repeat until the discovered model "looks plausible". Apart from wasted analyst hours, this trial-and-error procedure introduces subjectivity and the risk that an incorrect case notion propagates through subsequent dashboards, leading to wrong managerial decisions [3].

Additionally, two trends exacerbate this problem. First, the data deluge means that logs routinely exceed millions of events, where exhaustive, interactive inspection becomes infeasible. Second, PM is moving from expert-driven exploratory analyses towards continuous and citizen-developer scenarios, where domain specialists with limited PM expertise expect the tooling to "just work" [7]. In these settings, a robust, transparent and computationally lightweight way of detecting the case ID automatically is indispensable.

One of the primary challenges of case ID identification stems from the temporal and structural complexity of event logs. Attributes may exhibit similar patterns or distributions, making it difficult to distinguish the case ID from other attributes, such as resources or activities. In cyclic processes, where cases may restart or overlap, manually identifying the correct case ID becomes even more challenging.

The manual approach also introduces the risk of mislabelling, which can compromise the integrity of subsequent process mining analyses. Incorrectly identified case IDs lead to fragmented or erroneous process models, undermining the reliability of discovered insights. Furthermore, the time-intensive nature of manual labelling detracts from the overall efficiency of process analysis, particularly in real-time or high-frequency contexts where rapid decision-making is crucial [4].

Prior research has produced two main classes of approaches: (1) *Model-driven methods*, which evaluate every column (or combination of columns) as a tentative case ID, discover a process model and score the model by fitness or precision. Andaloussi et al. [8] pioneered this line with token-based replay, and later work refined the scoring techniques [2]. These methods are unsupervised and can, in principle, handle cyclic behaviour, but they are computationally heavy: discovering and replaying a model for each of n columns incurs $O(n)$ mining calls and becomes prohibitive on large logs; and (2) *Learning-based methods*, which train classifiers to recognise the "signature" of a case-ID column using engineered features such as uniqueness ratio, entropy or numeric patterns. Toyoda et al. [9] combined such a classifier with a second-stage miner to resolve ambiguities, and Sim et al. [10] use convolutional neural networks on image encodings. These methods can be highly accurate but depend on labelled corpora drawn from the same domain. Their performance may drop on unseen industries or when data protection rules prohibit log sharing.

Other lines of research address event-case correlation when the case ID is entirely absent [6,11,12]. They reconstruct traces by optimisation, sequence partitioning or probabilistic matching, often assuming a known process model or independence between parallel instances. While valuable, these techniques target a different problem formulation: they correlate events, not identify which column already contains the case ID.

This paper proposes a novel approach for identifying the case ID attribute in unlabelled event logs, based on a heuristic expression that measures the average time span of the values of each attribute in the event log.

The heuristic is designed for the *post-ingestion* stage, after the raw data have been flattened into a regular event log (one row = one event). It relies on two lightweight, yet essential, pre-conditions:

1. **At least one repeating categorical attribute.** A true case identifier is discrete and appears in multiple events. Hence the log must contain ≥ 1 string, code, or enumerated column whose values recur. If every column is either continuous (e.g. timestamps, amounts) or unique per event (e.g. UUIDs), automatic case-ID recovery is infeasible for any method.
2. **Single-process scope.** The log should already describe one coherent business process. Records from unrelated processes must be filtered out beforehand. Our method selects the case-ID column but does not decide what constitutes a process in the first place.

All 27 public logs used in our evaluation satisfy these assumptions, and they are typically met in industrial data preparations we have observed.

The proposed approach is intrinsically explainable and only requires the event log as input to be applied, being completely agnostic regarding the cyclic/acyclic nature of the business process subjacent to the event log.

This research seeks to contribute to the development of an automated and reliable method that can aid process engineers in handling unlabelled event logs more effectively. This method can then be implemented, for instance, as a software library, to be integrated in process mining tools, to help process engineers in this task. By systematically analysing temporal patterns and leveraging objective metrics, the proposed heuristic eliminates much of the guesswork and subjectivity inherent in manual methods. This not only improves accuracy and consistency but also significantly reduces the time and expertise required, enabling process engineers to focus on higher-value analytical tasks. Crucially, the heuristic needs no training data or process discovery, runs in $O(nlogn)$ time dominated by sorting, taking milliseconds for typical logs, and is intrinsically explainable: the chosen column is simply the one whose values occupy the narrowest temporal footprint.

The paper is structured as follows: Section 2 provides background information and reviews of related work. Section 3 describes the proposed method in detail. Section 4 presents the results and a case study dataset that verifies the effectiveness of the proposed method through various executions of event logs. Finally, in Section 5, we draw some conclusions and envisage future work.

2. Background and related work

In this section, we provide an overview of the background and related work in the field of process mining and the identification of the case ID attribute in event logs.

2.1. Process mining

Process Mining (PM) is a research field between data mining and Business Process Management (BPM) and analysis that aims to improve

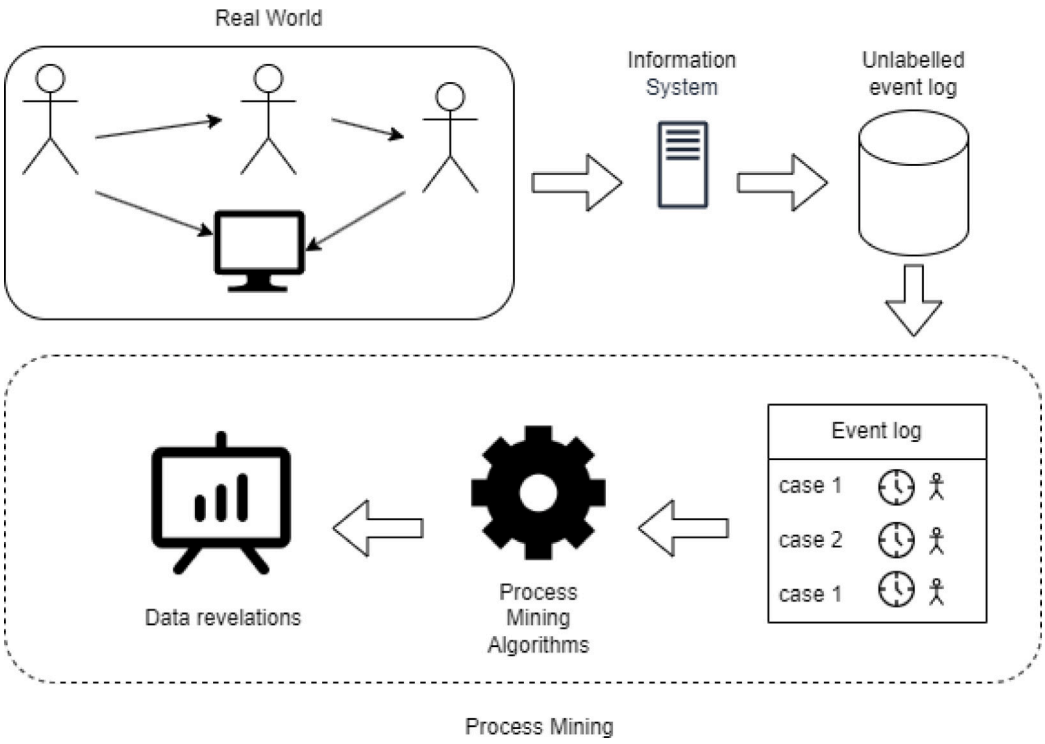

Fig. 1. Process Mining overview.

operational processes using event data [13]. The method begins by extracting data from databases in the form of event logs. These event logs serve as input for PM algorithms, and the output is then analysed. The results are presented visually through graphical images, as depicted in Fig. 1. This figure presents a portrait of the real world, where human or machine interactions take place within an environment. Information systems capture these interactions, giving rise to unlabelled event logs. These event logs serve as the foundation for PM. They flow through a sequence involving PM algorithms, leading to the revelation of crucial insights from data. This method facilitates the acquisition of key business process perspectives, including performance, data, organisational (resources), and control-flow.

The yielded results extend their utility beyond their visualisation. They offer insights, identifying bottlenecks and deviations, and even anticipating and diagnosing performance and compliance issues. This approach is adaptable across a broad spectrum of organisations and industries, underlining the wide applicability of PM principles.

2.2. Event logs

An event log is a record in the form of a list of events describing all the steps required during the execution of a business process. To be effective, each line of the log should contain certain key pieces of information, including a unique case ID, a label describing the event or activity being recorded, a start time indicating when the event began, and an end time indicating when the event ended. These four attributes are essential for accurately and effectively tracking and analysing the events recorded in the log. Other attributes can be used as well, such as for example, the resource(s) used to perform the activity (including rooms, machines and/or human resources), the cost of that activity, or any other data-related attribute to be further analysed.

The standard format for an event log is the eXtensible Event Stream (XES) format, which is supported by most PM tools [5]. In between the extraction of data from a data source and the creation of an event log in the XES format, it is common to use the Comma-Separated Values (CSV) as an intermediate format. It is used for storing tabular data in which the first row usually contains the names of each attribute, and the following rows are for values. The values in the fields are separated by commas. CSV files are commonly used for storing data from spreadsheets and databases and are often used for importing and exporting data between different information systems.

2.3. Related work

The problem of automatically identifying case-ID attributes in event logs has been approached from several angles. Early methods assume the existence of process models or structures and apply probabilistic or statistical techniques. For example, Ferreira and Gillblad [6] propose a Hidden Markov Model approach (Expectation–Maximisation) to correlate events into cases, and Walicki and Ferreira [11] use sequence partitioning (Markov chains) to detect event-case correlations. These model-based methods require prior knowledge (e.g., a process model or transition constraints) and are explicitly designed for acyclic processes, so they generally cannot handle loops or concurrency. This limits their generality and scalability to more complex workflows.

Building on this, Bayomie et al. [14] developed decision-tree and probabilistic models to infer case IDs. In their Deduce Case IDs (DCI) approach, Bayomie et al. use a decision-tree built from an unlabelled log and known process model to hypothesise possible case labels for each event. This approach requires as input both the process model and heuristic execution-time information; it outputs a (possibly probabilistic) labelling for each event. Bayomie et al. [12](2016) extend DCI to cyclic processes (DCIc) by adding a preprocessing step on the process model to encode loop behaviour, allowing case correlation in cyclic workflows. In all cases, these methods are computationally intensive (they often enumerate many label combinations) and their performance depends on the quality of the given model. For instance, Bayomie et al. [12] report ranking multiple candidate logs rather than a single solution. A later probabilistic approach by Bayomie et al. [15] also relies on a known model structure and infers case labels via Bayesian networks, again limiting interpretability. The authors in [16] take a pattern-mining approach: they subdivide the log, detect recurring

sequences of events, and then stitch them into traces. This three-step method can handle cyclic behaviour, but it presumes that repeated patterns uniquely identify cases, which may not hold in noisier settings.

Another category of methods treats case-ID identification as a pattern or heuristic discovery problem without requiring a full model. Andaloussi et al. [8] propose the Infer Case Id (ICI) method, which heuristically estimates the number of distinct values per attribute and then "labels" each candidate attribute by discovering a process model and computing control-flow quality metrics (fitness, precision, generalisation, simplicity). The attribute whose induced model scores highest is chosen as the case ID. This approach is conceptually simple and produces interpretable scores, and Andaloussi et al. report high accuracy on several real logs. However, it requires running a discovery algorithm for each candidate attribute (often using a fixed miner and a sample of the log), making it computationally expensive and sensitive to the choice of process-mining techniques. In fact, the authors note that their evaluation – though promising – is limited in scope and would benefit from testing with additional miners and logs.

Burattin & Vigo [17] take an even lighter-weight heuristic stance. Their chain-filtering algorithm scans the log for attributes whose values form "chains" of equality across successive events — in other words, a value that re-appears in temporally contiguous events is deemed a strong candidate for a case key. The attribute that yields the longest, most contiguous chains is selected as the case ID. The method is computationally inexpensive (essentially a single pass plus counting), requires no process model, and is fully interpretable: the output can be justified by showing the longest chains to an analyst. Its limitations are two-fold: (i) it implicitly assumes that case-ID values re-occur in an unbroken temporal block (no interleaving of cases), which is violated in logs with high concurrency, and (ii) it cannot handle continuous attributes or attributes that seldom repeat. In the authors' tests on document-management logs the heuristic was effective, but they did not report a general accuracy measure across multiple datasets.

Similarly, in [10] the authors apply a heuristic image-based approach: they convert each attribute's data into an "event density" image and train a Convolutional Neural Network (CNN) to classify attribute types (case ID, activity, etc.). This ML-driven method achieves very high overall accuracy (above 92%) on benchmark logs, but it requires a large labelled training set and is limited to the attribute types seen during training. Moreover, as noted by Brzychczy et al. [18], such classification models are black-boxes, lack guarantees beyond the chosen classes, and need manual filtering of unrelated attributes to avoid misclassification.

More recently, Toyoda et al. [9] propose a hybrid two-stage ML approach. In the first stage they use supervised learning to shortlist candidate columns for each key role (case ID, activity, timestamp), then in the second stage they exhaustively combine candidates and evaluate them by discovering process models and scoring them. This method significantly reduces combinatorial cost (from $O(n^3)$ to $O(k^3)$ where k is the number of candidates retained). The authors evaluate their method on 14 public event logs (BPI Challenge 2011–2020) and report that it correctly identifies the case-ID with about 71% accuracy (averaged over datasets). While this two-stage approach is effective on labelled business logs, it still relies on using process discovery and (implicitly) assumes the data are discrete event logs. Its ML component also requires labelled examples for each key attribute.

In the realm of continuous industrial process monitoring, researchers have developed methods to infer process instance boundaries (case IDs) from unlabelled sensor event streams. Helal et al. [19] present a real-time CEP-based technique that correlates raw IoT events into cases on the fly, showing competitive accuracy on live sensor data and outperforming baseline methods in throughput. Bayomie et al. [20] propose a probabilistic event-correlation approach that uses process model constraints to split continuous event sequences into traces, achieving F1-scores up to ≈92.5% in correctly segmenting cases on real-world logs. Similarly, Brzychczy et al. [18] devise a rule-based algorithm for cycle detection in mining sensor data, which identifies case boundaries by detecting significant changes in time-series signals. Their method was validated on both synthetic and industrial datasets, yielding F1 scores of ≈96%–97% on mining equipment logs and ≈92.6% on manufacturing sensor data.

These works provide evidence that automated trace segmentation from continuous signals is feasible, with high accuracy, in practical process mining scenarios. While these methods focus on the upstream task of deriving structured event logs from raw continuous data (e.g., via CEP-based streaming segmentation, probabilistic event-case correlation, or rule-based cycle detection in time-series data), our heuristic addresses the subsequent step in the pipeline. These existing approaches transform unlabelled sensor or time-series streams into event logs by correlating events into process instances — a necessary precursor to any process mining analysis. In contrast, our proposed heuristic assumes that such an event log has already been obtained and operates after this segmentation/correlation step, automatically selecting the most likely case-ID attribute from the resulting log. In this way, it complements prior correlation methods as a lightweight post-processing technique: once continuous data has been converted into an event log, our method can swiftly identify the case identifier. Moreover, the heuristic is deliberately lightweight and domain-agnostic — it requires no predefined process model, no complex CEP engine or rules, and no labelled training data. Instead, it leverages only the temporal properties of the event data to rank candidate attributes, running in milliseconds of computation. This makes our approach an efficient add-on to any continuous-data event correlation pipeline, providing quick case-ID labelling without the overhead of more involved modelling or training steps.

In summary, existing approaches to case-ID detection trade off different strengths and limitations. Model-based and tree-based methods (Ferreira et al., Walicki et al., Bayomie et al.) can leverage control-flow structure but require known models and do not scale well to large attribute sets or noisy data. Heuristic methods (Andaloussi et al., Sim et al.) need no training but often need many discovery steps or manual tuning of parameters, and may implicitly assume finite sets of attribute types or log maturity. ML approaches (Toyoda et al., Sim et al.) can automate detection with fewer assumptions about domain, but they need training data and their outputs can be hard to interpret. Crucially, none of the above (aside from Brzychczy et al.) are designed for continuous sensor streams. As a result, most prior methods offer limited scalability or generality: for example, they cannot easily incorporate new events or attributes without re-running costly discovery, nor do they explain their decisions in human-understandable terms.

3. Method

This section formalises the proposed method. It first specifies a log-independent normalisation procedure that converts any raw dataset into a canonical event-log representation; the same procedure is applied to every dataset. It then derives the heuristic score h_{attr} and justifies the default exponent parameters (a, b, c). All matters pertaining to empirical validation – including the composition of the BPI-Challenge datasets, the experimental protocol, and the performance metrics – are deferred to Section 4.

3.1. Input normalisation (log-independent)

The data input normalisation step was divided into three phases: event log cleaning, temporal columns identification, and column preparation (each column corresponds to an event log attribute). The event log cleaning step began with the removal of date columns to streamline the dataset, followed by the elimination of empty columns and lines with null values. Columns with only one or two unique values were excluded, as well as columns with only distinct values (where no value occurs more than once). Finally, any columns found to be redundant

Table 1
Literature overview of process-instance case-ID identification. Figures are quoted verbatim from their sources and are *not* directly comparable. "Categorical attributes" means the method expects discrete column values (codes, strings). "Training?" notes whether labelled traces are required; "Cyclic?" whether loops are handled. Accuracy and runtime are those reported by the original authors; "n/a" indicates none given. Interpretability is a qualitative tag (high for rule/DT models, low for complex ML). A uniform benchmark with common data and metrics appears in Table 5.

Reference	Approach	Training?	Data	Cyclic?	Complexity	Runtime	Accuracy (reported)	Interpretability
Ferreira & Gillblad [6]	Model-based (EM)	Yes	Discrete event types (Markov)	No	High	Sec. to min.	Up to 98% G-score on synthetic logs	Moderate (probabilistic)
Walicki & Ferreira [11]	Model-based (partition)	No	Symbolic sequences	Yes	Very High	min. to hours	Exact solutions found; minimal pattern sets	High (algorithmic)
Bayomie et al. [12]	Model-based (alignment)	No	Timestamped events	Yes	High	35 min-11 h	Precision: 47%–82%, Recall:47-80%	High (tree structure, ranked logs, model-based)
Andaloussi et al. [8]	Heuristic (discovery-based)	No	Discrete event attributes	No	Moderate	min. per log	81%–94% (depending on log and miner)	High (score-based)
Burattin & Vigo [17]	Heuristic (chain filtering)	No	Decorative attributes	No	Moderate	Sec. to min.	Validated by domain experts, not quantified	High (simple rules)
Lichtenstein et al. [3]	Attribute-driven ML-based heuristic	Yes	Discrete event types, categorical attributes	Yes	High	20 s for small datasets	Gsim-score: up to 80%	High (attribute weights and trace generation)
Sim et al. [10]	Deep CNN	Yes	Mixed categorical and numerical attributes	Yes	High	Slow (depends on log size and GPU)	Up to 97.8%	High (embedding visualisations and quality metrics)
Toyoda et al. [9]	ML-based (classification + miner)	Yes	Discrete sequences, categorical attributes	Yes	Moderate ($O(k^3)$)	20 s for small datasets	Up to 80% (key attribute id)	Moderate (two-stage)
Brzychczy et al. [18]	Heuristic (pattern rules)	No	Mixed continuous/binary signals	Yes	Moderate	Sec. per dataset	F1 ≈ 96.8-97.0% (mining), 92.6% (mfg.)	High (rule-based)
This paper	Heuristic (temporal-span)	No	Discrete event types, categorical attributes	Yes	Low (single scan + sort)	Millisec. to sec.	85.2% (top-1), 96% (top-2)	High (explainable rule)

(carrying the same information for every event) were merged into a single attribute.

For temporal columns identification, an event log may store temporal information in one of two common ways: (i) a single timestamp per event, or (ii) two distinct columns that record the start and the completion of an activity. Our normalisation routine first detects every column that parses as a temporal value and then, if two such columns exist, verifies a start/complete relationship by checking that the first timestamp precedes the second. In the 27 BPI-Challenge logs analysed later (Section 4), 13 logs expose a start/complete pair and the routine labelled both columns correctly in all of them; the remaining logs contain only one timestamp, which is simply taken as the event time. Consequently, the heuristic can operate unchanged regardless of whether a log provides one or two timestamp columns.

The column preparation phase was applied as follows: for each attribute of the event logs used to assess our proposal, a new dataset was built containing just the start date and the column corresponding to the attribute under assessment. Each one of these datasets were used, both to build the corresponding dispersion chart (see Fig. 2) and to enable the application of the proposed heuristic. That is, the heuristic is applied to each one of these datasets separately. Each dataset is first sorted by start date and then by the values of the target attribute so that the lines having the same attribute value are grouped together and remain chronologically ordered.

For the data exploration step, we analysed the event logs using techniques that include the visualisation of frequency charts, box plots, column analysis, and assessment of missing and unique values. To gain insight on what differentiates the case ID attribute from other attributes, we additionally generated dispersion charts (Fig. 2), each one depicting data for a given attribute of an event log. The vertical axis represents all the distinct values of the attribute whose data is being depicted in the chart, ordered, from bottom to top, by the start date of the first occurrence of each value (red dots in Fig. 2). The

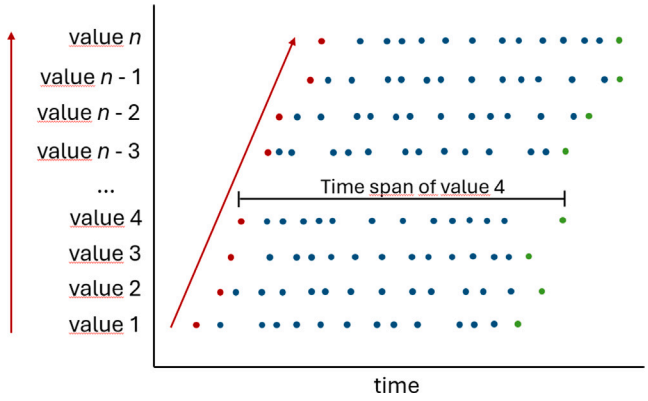

Fig. 2. Schema of a dispersion chart.

horizontal axis represents time. This means that the points in each line in a chart correspond to a specific value of the event log attribute the chart stands for, and each point in the line represents the moment in time at which some event with that attribute value has occurred. That is, each dispersion chart depicts the temporal distribution of the occurrences of the values of an attribute. Finally, we define the time span of some value as the difference between the start date of the last occurrence of that value (green dots) and the start date of the first occurrence (red dots).

Fig. 3 displays temporal-dispersion charts for six individual attributes taken from a representative event log. Chart A corresponds to the ground-truth *case-ID* column. Panels B to F show other attributes whose business labels are immaterial to the methodological point being illustrated.

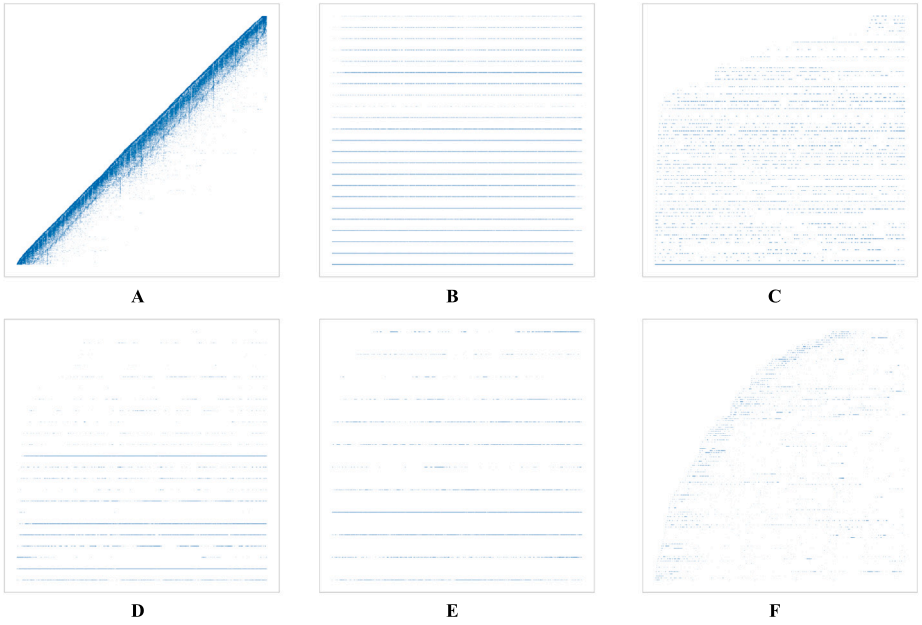

Fig. 3. Examples of dispersion charts showing the time span of the values of event log attributes. Note: the axes labels are not shown (too many) for clarity. Chart A shows the case-ID column; Charts B–F show other attributes from the same log.

Two visual cues separate Chart A from the others: first, its values form a clean diagonal, indicating that each case occupies a short, non-overlapping time interval; second, the per-value spans in Chart A are markedly shorter than those in panels B–F. These characteristics – tight, non-overlapping spans and a diagonal footprint – are precisely the patterns quantified by the span-ratio and coverage terms in our proposed heuristics.

Fig. 4 shows four more typical examples of case ID dispersion charts following a similar pattern. While this pattern is not universal (we discuss two such cases in Section 4), it can be explained by the fact that business process cases usually have a well-defined start and end dates while, for example, activities tend to appear throughout all the time span of an event log, dispersed throughout many cases/processes. This analysis led us to formulate the hypothesis that the average time span of the values in the case ID attribute is smaller than for other attributes in the event log. This insight was fundamental in developing the heuristic expression outlined in the next section, which we use to identify the case ID attribute.

3.2. Proposed heuristic

The proposed heuristic for identifying the case ID attribute within unlabelled event logs is based on the hypothesis, posed in the data exploration phase, that, on average, the values of the case ID attribute exhibit a shorter time span compared to the values of other attributes. This means that the proposed heuristic needs to measure the average time span for the values of each attribute. Now, each attribute has different values and the number of occurrences of each value varies from value to value. That is, some values occur more often than others. So, instead of a simple average expression, we propose a heuristic consisting of a weighted average of values' times spans where the time span for each value is weighted by the number of occurrences of that value. The idea is that the proposed heuristic effectively captures the temporal patterns that differentiate the case ID attribute from others in the event log dataset. The heuristic expression is applied to each attribute values at a time, and it is defined as follows:

$$h_{attr} = \frac{\sum_i [(TMax_i - TMin_i)^a \cdot NO_i]^b}{NUV^c} \quad (1)$$

where, h_{attr} is the heuristic value computed for attribute attr, $TMax_i$ and $TMin_i$ are, respectively, the maximum and minimum start date values for the value with index i in the attribute's column, NO_i is the number of occurrences of a value in the attribute's column, NUV is the number of unique values of the attribute, the a, b, and c exponents are parameters that can be adjusted to fine-tune the heuristic performance with the idea of changing the importance of each element in the expression.

The higher parameter a, the higher the importance of the values' time span regarding the number of occurrences of the values; The higher parameter b, the higher the importance of the weighted sum of the time spans regarding the number of unique values; The higher parameter c, the higher the importance of the number of unique values regarding the weighted sum of the time spans.

Given some event log, Eq. (1) is separately applied to all its attributes. The attribute with the smallest h_{attr} value is considered to be the case ID attribute. However, in a real situation we can think of a PM tool software feature where, let us say, the two or three attributes with the smaller values are presented to the user, who can then take a final decision about which one of these corresponds to the case ID attribute. We should also clarify that, while the insight that led to the formulation of this expression came from the analysis of dispersion charts, where most of the case ID charts have a diagonal shape, the proposed heuristic expression only accounts for the average time span of the attributes' values, not the diagonal shape formed by the events. Finally, it is worth mentioning that the results returned by the approach are intrinsically explainable as the attribute identified as case ID is always the one whose values have the smallest average time span.

Therefore, the design of this heuristic expression follows three pragmatic principles:

1. *Orthogonality:* three factors capture distinct evidence that an attribute is a case identifier: (i) short temporal span per value ($TMax - TMin$), (ii) sufficient repetition-measured by the occurrence count NO_i, and (iii) moderate overall variety, penalised through the number of unique values NUV.
2. *Scale-robust weighting:* exponents $a, b, c \geq 0$ act as unit-free weights, allowing analysts to increase or decrease each cue's influence without introducing additional parameters or changing the units of measurement.

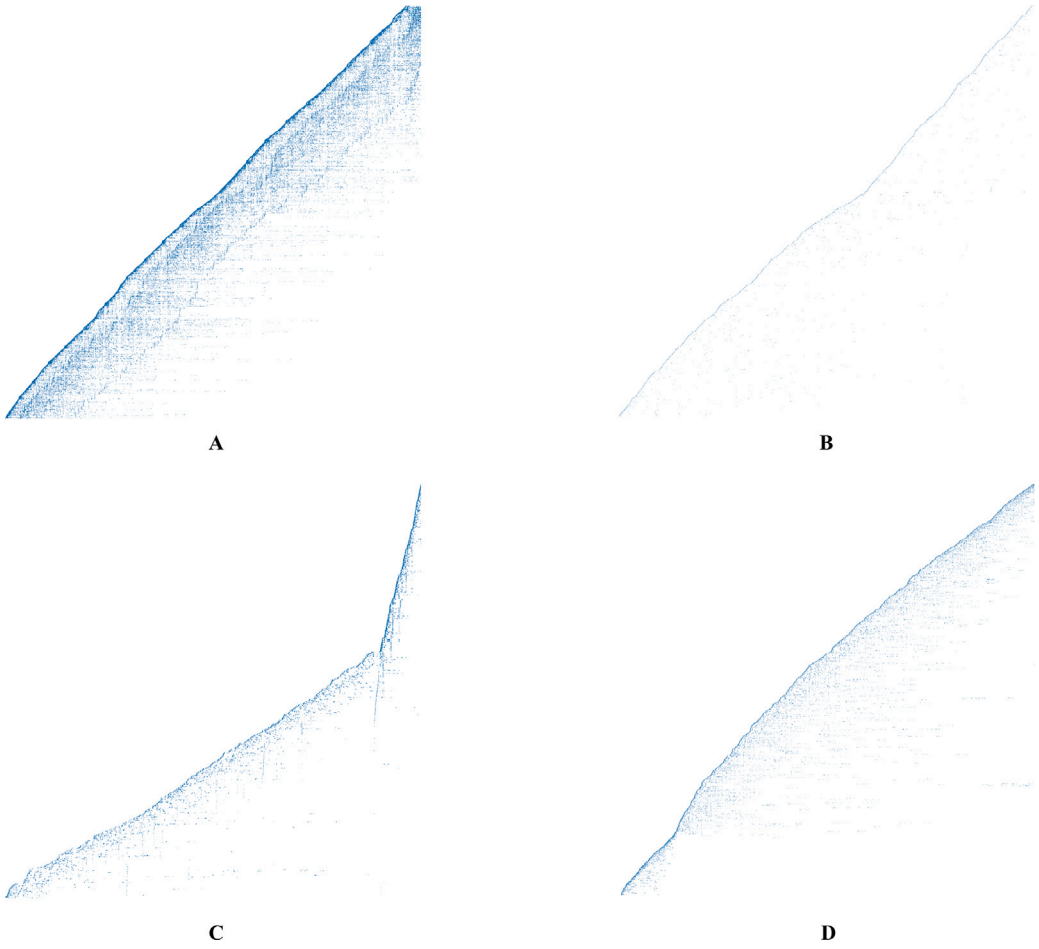

Fig. 4. Examples of dispersion charts for case ID attributes of different event logs, following a diagonal pattern.

3. *Parsimony:* a single multiplicative score with exactly three tunable numbers keeps the method transparent and easy to configure. The default triplet $(a, b, c) = (0.5, 1, 2)$ was chosen after exploratory tuning on a diverse set of public logs; simpler alternatives such as an unweighted sum or a geometric mean behaved noticeably less reliably in those trials. An extensive grid search revealed a broad plateau in which hit-rate varies by less than 1%. We adopt the plateau centre $(a, b, c) = (0.5, 1, 2)$ as the recommended default; exploration of automatic fine-tuning strategies is left for follow-up work".

Regarding the computational complexity of the approach, there are three sequential steps to consider: sorting the event log by the start date attribute, grouping the events by attribute value, and the application of the heuristic expression to each attribute. The sorting process uses the QuickSort algorithm, which has complexity $O(n \log n)$, where n, in this case, is the number of events in the event log. The grouping process has complexity $O(n)$, since it needs just a single iteration over the event log and the insertion of events in their groups takes $O(1)$ if we use a hash table. Finally, the application of the heuristic expression has complexity $O(N\_attr \times max\_values)$, where N_attr is the number of attributes of the event log, and max_values is the number of values of the attribute with the larger number of values in the event log. Given that $n \gg N\_attr$ and $n \gg max\_values$, $O(n \log n)$ will in general be larger than $O(N\_attr \times max\_values)$ as n grows. So, we can conclude that the computational complexity of the proposed approach is $O(n \log n)$. In practical terms, each one of the event logs was processed in less than a minute, and the most part of them, in just a few seconds. The hardware configuration used was the following: CPU – i7; RAM: 16 GB; GPU: NVIDIA. Finally, the proposed method does not need a training phase as in [9,10], as well as a previous manual labelling of the attributes for this purpose.

4. Results and discussion

4.1. Datasets used

The dataset used to test the proposed heuristic is composed by the 27 available event logs from the Business Process Intelligence (BPI) Challenge, spanning from 2011 to 2020. Table 2 summarises the event logs used, including their brief descriptions, number of attributes and number of rows.

These real event logs exhibit significant variety. Regarding the business area, they cover areas as health treatments processes in a hospital, application processes for personal loans, incident management systems, application processes for construction permits, and customer interaction in the insurance area, to name just a few. Regarding the number of attributes, the event log with less attributes has 6 attributes and the one with more attributes has 174 attributes. The average number of attributes is about 31. Regarding the number of events/rows, the event log with less events has 289 events and the one with more events has 9 329 418 events. The average number of events is about 854 437.

4.2. Applying the heuristics

To explore the effectiveness of the heuristic, we conducted several experiments using different combinations of the a, b, and c parameter

Table 2
Summary description of the event logs used to test our proposed heuristic.

Event log	Description	Attributes	Rows
BPI 2011 [21]	Logs of a Dutch academic hospital	128	150,291
BPI 2012 [22]	Logs of a loan application process	6	262,201
BPI 2013a [23]	Logs of Volvo IT incident and problem management, closed problems	13	6660
BPI 2013b [24]	Logs of Volvo IT incident and problem management, incidents	13	65,533
BPI 2013c [25]	Logs of Volvo IT incident and problem management, open problems	12	2351
BPI 2014a [26]	Rabobank Group activity log for incidents	7	466,737
BPI 2014b [27]	Rabobank Group change details	21	30,275
BPI 2014c [28]	Rabobank Group incident details	28	46,606
BPI 2014d [29]	Rabobank Group interaction details	17	147,004
BPI 2015a [30]	Logs of five Dutch municipalities, municipality 1	29	52,217
BPI 2015b [31]	Logs of five Dutch municipalities, municipality 2	28	44,354
BPI 2015c [32]	Logs of five Dutch municipalities, municipality 3	29	59,681
BPI 2015d [33]	Logs of five Dutch municipalities, municipality 4	29	47,293
BPI 2015e [34]	Logs of five Dutch municipalities, municipality 5	29	59,083
BPI 2016a [35]	Employee Insurance Agency - clicks logged in	20	7,174,934
BPI 2016b [36]	Employee Insurance Agency - clicks not logged in	15	9,329,418
BPI 2016c [37]	Employee Insurance Agency - complaints	18	289
BPI 2016d [38]	Employee Insurance Agency - questions	17	123,403
BPI 2016e [39]	Employee Insurance Agency - Werkmap messages	8	66,058
BPI 2017 [40]	Logs of a loan application process of a Dutch financial institute	19	561,671
BPI 2018 [41]	Logs of applications for EU direct payments handling	75	2,514,266
BPI 2019 [42]	Logs from a multinational company operating in the area of coatings and paints	22	1,595,923
BPI 2020a [43]	Logs from a university for domestic travel expense claims	11	56,437
BPI 2020b [44]	Logs from a university for international travel expense claims	24	72,151
BPI 2020c [45]	Logs from a university for prepaid travel expense claims	23	18,246
BPI 2020d [46]	Logs from a university for requests for payment (not travel related)	15	36,796
BPI 2020e [47]	Logs from a university for travel permits	174	86,581

Table 3
Heuristic configurations used in the experiments.

Heuristic	**a**	**b**	**c**
A	1	1	1
B	1	1	2
C	2	1	1
D	1	2	1
E	1/2	1	2
F	1	1/2	2
G	1/2	1	1
H	1	1/2	1

values of the heuristic expression. The eight better configurations, whose results we present below, are shown in Table 3.

We applied the heuristic with the above configurations to assess its effectiveness in identifying the case ID attribute. Table 4 presents, for each testing event log, the corresponding position rank assigned to the case ID attribute for each heuristic configuration (columns with headers from A to H): rank 1 means that, among all attributes, the case ID attribute has received the best rank (the smaller value computed with the heuristic expression for all attributes of the event log), rank 2 means that it received the second-best value, and so on. For example, in the BPI Challenge 2015b event log, configuration A resulted in a rank of 2, while configuration B obtained a rank of 1. The numbers between parenthesis displayed for event log BPI Challenge 2018 correspond to the number of attributes that are on the same rank as the case ID attribute. The $NAttrs$ column represents the number of attributes in each event log that were considered for case ID identification after the data preparation step. This count includes all the potential candidate attributes evaluated by the different heuristic's configurations. The number in this column provides an understanding of the event log's complexity and the variety of attributes that were assessed during the case ID identification process. Furthermore, the Total row in Table 4 summarises the overall performance of each configuration across all event logs. The fractions indicate the number of event logs for which each configuration achieved the top rank, out of the total number of event logs used in the experiments.

Analysing columns A to H from Table 4, it becomes evident that configuration E consistently outperformed other configurations by achieving the highest rank for most event logs. This configuration assigned rank 1 to the case ID attribute in 23 out of 27 event logs, corresponding to an 85.2% success rate. We can also comment on the robustness of the heuristic to changes in the a, b, and c parameters: two of the configurations have a success rate larger than 81.5% and five have a success rate between 74.1% and 77.8%. Furthermore, for the most part of the situations where the case ID attribute is not ranked first, it is ranked second. In fact, in a scenario where we wanted to present the user the two best ranked attributes, if we used configuration E, the case ID attribute would be presented to the user for 96% (26 out of 27) of the tested event logs. In the same scenario but using configuration H (the one achieving the worst results), the case ID attribute would be presented to the user for 89% of the event logs. If, instead, we presented the three best ranked attributes, the case ID attribute would be presented for 100% of the tested event logs, if we used configuration E, and 96%, if we used configuration H.

For the same event logs set, configuration E of the proposed heuristic only misses the identification of the case ID attribute for the BPI Challenge 2018. That is, it correctly assigns rank 1 to the case ID attribute 13 out of 14 times, which corresponds to a success rate of 92.86%. When comparing to Toyoda et al. [9], only configuration H has an accuracy below the announced 71.43% on these event logs, which means that all the other 7 configurations achieve an equal or higher performance.

We now analyse the case of event log BPI Challenge 2011. This an event log that covers events spanning from January 2005 to March 2008 regarding the diagnosis and treatments performed on patients in the Gynaecology department of a Dutch Academic Hospital. As we can see in Fig. 5A, the dispersion chart of the case ID attribute does not follow the thin diagonal pattern observed for the most part of the remaining testing event logs. We speculate that this is due to fact of the time span of this event log being smaller regarding the typical time span of the processes in this area. Medical monitoring of one patient may last for many years, especially in the case of chronical diseases.

The near triangular shape of the dispersion chart just indicates that many of the processes in this period just did not finish during

Table 4
Results obtained for different configurations of the heuristic expression on the testing BPI Challenge event logs. Columns with headers A–H show the rank obtained by the case ID attribute for the eight tested configurations of the heuristic. The last column N Attr shows the number of columns of each event log after the data cleaning process.

Event log	A	B	C	D	E	F	G	H	N Attrs
BPI 2011	2	2	2	2	1	1	2	2	54
BPI 2012	1	1	1	1	1	1	1	1	4
BPI 2013a	1	1	1	1	1	1	1	1	11
BPI 2013b	1	1	1	1	1	1	1	1	11
BPI 2013c	1	1	1	1	1	1	1	1	10
BPI 2014a	1	1	1	1	1	1	1	1	5
BPI 2014b	1	1	1	1	1	1	1	1	9
BPI 2014c	2	2	2	2	2	2	2	2	22
BPI 2014d	2	2	2	3	2	2	2	2	12
BPI 2015a	1	1	2	2	1	1	1	1	17
BPI 2015b	2	1	3	3	1	1	2	2	16
BPI 2015c	1	1	1	1	1	1	1	2	16
BPI 2015d	1	1	1	1	1	1	1	2	17
BPI 2015e	1	1	1	1	1	2	1	2	18
BPI 2016a	3	2	3	3	2	2	3	5	12
BPI 2016b	1	1	1	1	1	1	1	1	9
BPI 2016c	1	1	1	1	1	1	1	1	9
BPI 2016d	1	1	1	1	1	1	1	1	12
BPI 2016e	1	1	1	1	1	1	1	1	4
BPI 2017	1	1	1	1	1	2	1	2	15
BPI 2018	3(2)	3(2)	3(2)	3(2)	3(2)	3(2)	3(2)	3(2)	27
BPI 2019	1	1	1	1	1	1	1	1	14
BPI 2020a	1	1	1	1	1	1	1	1	5
BPI 2020b	1	1	1	1	1	1	1	1	17
BPI 2020c	1	1	1	1	1	1	1	1	17
BPI 2020d	1	1	1	1	1	1	1	1	9
BPI 2020e	1	1	1	1	1	1	1	3	58
Total:	21/27	22/27	20/27	20/27	**23/27**	21/27	21/27	16/27	
Success Rate:	77.8%	81.5%	74.1%	74.1%	**85.2%**	77.8%	77.8%	59.3%	

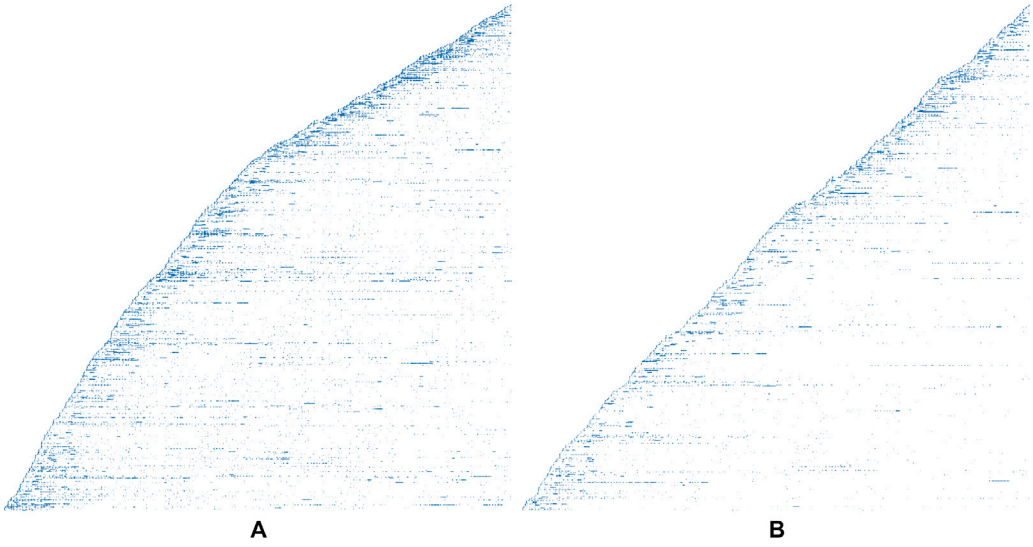

Fig. 5. BPI Challenge 2011 dispersion charts for (A) the case ID attribute and (B) the diagnosis treatment combination ID attribute.

the period covered by the event log. Even so, configurations E and F of the proposed heuristic managed to assign rank 1 to the case ID attribute, while the remaining configurations assigned it rank 2 (Fig. 5B shows the dispersion chart for the diagnosis treatment combination ID attribute, which ranked first for these configurations). This can be explained due to the fact that configurations E and F use value $\frac{1}{2}$, respectively, for parameters a and b of the heuristic expression, thus diminishing the magnitude of the time span of the attribute values (the numerator in Eq. (1)) when compared to the number of unique values of the case ID attribute (the denominator in Formula (1)) in the heuristic expression.

Another case that is worth of discussion is the case of BPI Challenge 2018, for which the case ID attribute is ranked third for all configurations, therefore, being one of the event logs with worst results. When we apply the heuristic to this event log, attributes *docid* and *docid_uuid* share the top position. This happens because these two attributes exhibit a near-perfect correlation. The *docid* attribute stands for the internal identifier of the document related to the event, while *docid_uuid* represents a globally unique identifier for the document to which the event is associated. Fig. 5 depicts dispersion charts for these two attributes (charts B and C for the *docid* and *docid_uuid* attributes, respectively) and for the case ID attribute (chart A). As described

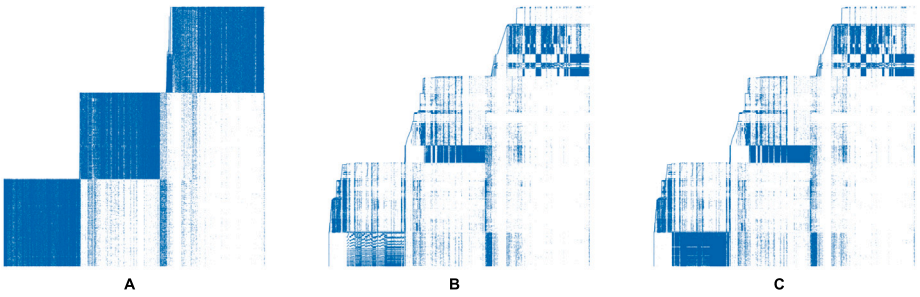

Fig. 6. Dispersion charts of the BPI Challenge 2018 three top attributes: A — case ID (rank 3); B — docid attribute (rank 1); C — docid_uuid attribute (rank 1).

Table 5
Quantitative comparison of methods whose goal matches ours: *selecting* the existing case-ID column in a flat event log. Accuracy = share of logs where the correct column is ranked first. Wall-clock runtimes are minima–maxima per log. ‡ = measured on an Intel i7-11700 (8× 2.5 GHz, 16 GB RAM, SSD).

Reference	Approach	Accuracy	Runtime	Datasets
Andaloussi et al. [8]	Heuristics (discovery-based)	81%–94% (depending on log and miner)	Minutes per log	4 BPI logs
Toyoda et al. [9]	ML-based (classification + miner)	~70% average (key attribute id)	15-42 s	14 BPI logs
This work	Span-Heuristic	85.2% (top-1), 96% (top-2)	0.02 to 0.20 s‡	27 BPI logs

in the dataset documentation [41], "this dataset covers the handling of applications for EU direct payments for German farmers from the European Agricultural Guarantee Fund". This process is repeated every year with minor changes. Chart A in Fig. 6 reflects that clearly: each year, all the cases start nearly at the same time and the main activity level for each case happens during the first year of the process, which is visible through the three well defined dense blocks. However, we see that the events related to some cases keep appearing beyond the first year, although with less intensity. This pattern differs from the "diagonal band" pattern shown in Figs. 3A and Fig. 4, observed for the most part of case ID attributes. This distinct behaviour is enough to lengthen the time span of the case ID values and explains why, for this dataset, the case ID attribute is not ranked first. In situations like this one, the analysis of dispersion charts may also help an expert in the area identifying the attribute corresponding to the case ID among the best ranked attributes.

Among the studies presented previously in Table 1, only two approaches satisfy requirements for a fair comparison, namely:

1. they pursue the same objective-selecting an already-present column as the case identifier;
2. they publish (or allow us to reproduce) quantitative accuracy;
3. they provide at least one wall-clock runtime figure;
4. they use the same datasets (BPI Challenge).

These approaches are the ones from Toyoda et al. [9] and Andaloussi et al. [8]. Their headline metrics are collected in Table 5.

Our heuristic matches the best reported effectiveness while improving median runtime by two orders of magnitude (0.02 to 0.20 s per log) compared with [9], and by more than one order compared with [8]. Moreover, when the top-3 ranked attributes are shown to the analyst, the correct case-ID appears for all 27 public logs in our benchmark dataset. This supports the claim that lightweight statistical cues – specifically span ratio and value entropy – are sufficient for reliable case-ID selection in most real-world logs.

5. Conclusions and future work

In this work, we propose a heuristic approach for the identification of the case ID attribute in unlabelled event logs. The proposed heuristic is based on the hypothesis that the time span of each of the values of the case ID attribute is smaller on average when compared to the average time span of values of other attributes. The heuristic expression takes the form of a weighted average of the time spans of the attributes' values, with customisable parameters that allow for flexibility in assigning importance to its components. The method proposed only needs as input the event log data and it can be applied whether the process model of the event log is cyclic or not. It is also intrinsically explainable: the attribute identified as case ID is the one whose values have the smallest time span on average.

The heuristic was applied to 27 event logs taken from the Business Process Intelligence (BPI) Challenge. The best performant configuration of this heuristic successfully identified the case ID attribute in 23 out of 27 event logs, achieving an 85,2% success rate. The results also show the robustness of the approach, with 7 out of 8 heuristic configurations achieving 74,1% success rate or more. Also, for the most part of the event logs and heuristic configurations, when the case ID attribute is not ranked first, it is ranked second. Finally, the best configuration of the proposed heuristic always assigns rank 1 to the case ID attribute in the event logs also tested by the two approaches with which it is possible to directly compare the proposed approach.

The results of this study demonstrate the potential of the proposed heuristic as a practical and efficient tool for identifying the case ID attribute in unlabelled event logs. The heuristic's simplicity and explainability set it apart from more complex, supervised approaches that often require pre-labelled data or intricate training pipelines. By focusing on temporal patterns and leveraging a weighted average approach, this method effectively isolates the case ID attribute with higher success rates across diverse real-world datasets. These findings underscore the heuristic's applicability to both cyclic and acyclic business processes, offering a robust solution for practitioners in process mining.

Despite its strengths, the heuristic has limitations, particularly in cases where attributes exhibit high correlation, as seen in the BPI Challenge 2018 dataset. This highlights the need for further refinement to handle multi-correlated data more effectively. Additionally, while the heuristic achieves a high rank for case IDs in most scenarios, there remain edge cases where expert intervention or complementary techniques may be required. Future work will address these challenges by exploring hybrid approaches that combine the heuristic with machine learning methods for greater adaptability.

Expanding the scope of the heuristic to identify other key attributes, such as activities and timestamps, represents another promising avenue for research. Incorporating natural language processing techniques could enhance the identification of text-based attributes, which are common in unstructured event logs. Furthermore, developing a comprehensive preprocessing pipeline that integrates the heuristic with other attribute detection methods could provide a holistic solution for managing unlabelled event logs.

Finally, we envision integrating the heuristic into widely used PM tools to streamline its adoption in industry settings. By automating the identification of case IDs and other critical attributes, this integration could significantly reduce the time and expertise required for preprocessing, making process mining more accessible and scalable. Such advancements will not only enhance operational efficiency but also open new possibilities for real-time process monitoring and improvement.

CRediT authorship contribution statement

André Vicente: Writing – original draft, Software, Investigation, Formal analysis, Data curation, Conceptualization. **Carlos Grilo:** Writing – review & editing, Writing – original draft, Validation, Supervision, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Rui Rijo:** Writing – review & editing, Writing – original draft, Validation, Supervision, Methodology, Investigation, Formal analysis, Conceptualization. **Ricardo Martinho:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Resources, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was financially supported by Project "ProM4Prod - Plataforma de Process Mining para descoberta, medição, monitorização e otimização de processos de produção, Portugal", CENTRO-01-0247-FEDER-047242, in the scope of Portugal 2020, cofunded by FEDER (Fundo Europeu de Desenvolvimento Regional) under the framework of PO CENTRO (Programa Operacional da Região Centro).

Data availability

Datasets used are publicly available.

References

[1] Diba K, Batoulis K, Weidlich M, Weske M. Extraction, correlation, and abstraction of event data for process mining. WIREs Data Min Knowl Discov 2020;10(3). http://dx.doi.org/10.1002/WIDM.1346.

[2] de Murillas EGL, Reijers HA, van der Aalst WMP. Case notion discovery and recommendation: automated event log building on databases. Knowl Inf Syst 2020;62(7):2539–75. http://dx.doi.org/10.1007/S10115-019-01430-6.

[3] Lichtenstein T, Bano D, Weske M. Attribute-driven case notion discovery for unlabeled event logs. In: Marrella A, Weber B, editors. Business process management workshops - BPM 2021 international workshops, Rome, Italy, September 6-10, 2021, revised selected papers. Lecture notes in business information processing, vol. 436, Springer; 2021, p. 111–22. http://dx.doi.org/10.1007/978-3-030-94343-1_9.

[4] Fazio RD, Balzanella A, Marrone S, Marulli F, Verde L, Reccia V, Valletta P. Caseid detection for process mining: A heuristic-based methodology. In: Smedt JD, Soffer P, editors. Process mining workshops - ICPM 2023 international workshops, Rome, Italy, October 23-27, 2023, revised selected papers. Lecture notes in business information processing, vol. 503, Springer; 2023, p. 45–57. http://dx.doi.org/10.1007/978-3-031-56107-8_4.

[5] van der Aalst WMP. Process Mining - Data Science in Action. second ed.. Springer; 2016, http://dx.doi.org/10.1007/978-3-662-49851-4,

[6] Ferreira DR, Gillblad D. Discovering process models from unlabelled event logs. In: Dayal U, Eder J, Koehler J, Reijers HA, editors. Business process management, 7th international conference, BPM 2009, Ulm, Germany, September 8-10, 2009. proceedings. Lecture notes in computer science, vol. 5701, Springer; 2009, p. 143–58. http://dx.doi.org/10.1007/978-3-642-03848-8_11.

[7] Geyer-Klingeberg J, Nakladal J, Baldauf F, Veit F. Process mining and robotic process automation: A perfect match. In: van der Aalst WMP, Casati F, Conforti R, de Leoni M, Dumas M, Kumar A, Mendling J, Nepal S, Pentland BT, Weber B, editors. Proceedings of the dissertation award, demonstration, and industrial track at BPM 2018 co-located with 16th international conference on business process management. CEUR workshop proceedings, vol. 2196, CEUR-WS.org; 2018, p. 124–31, URL: https://ceur-ws.org/Vol-2196/BPM_2018_paper_28.pdf.

[8] Andaloussi AA, Burattin A, Weber B. Toward an automated labeling of event log attributes. In: Gulden J, Reinhartz-Berger I, Schmidt R, Guerreiro S, Guédria W, Bera P, editors. Enterprise, business-process and information systems modeling - 19th international conference, BPMDS 2018, 23rd international conference, EMMSAD 2018, held at CAISE 2018, Tallinn, Estonia, June 11-12, 2018, proceedings. Lecture notes in business information processing, vol. 318, Springer; 2018, p. 82–96. http://dx.doi.org/10.1007/978-3-319-91704-7_6.

[9] Toyoda K, Ying RGK, Zhang AN, Siew TP. Identifying the key attributes in an unlabeled event log for automated process discovery. IEEE Trans Serv Comput 2024;17:74–81. http://dx.doi.org/10.1109/TSC.2023.3330175.

[10] Sim S, Sutrisnowati RA, Won S, Lee S, Bae H. Automatic conversion of event data to event logs using CNN and event density embedding. IEEE Access 2022;10:15994–6009. http://dx.doi.org/10.1109/ACCESS.2022.3143609.

[11] Walicki M, Ferreira DR. Sequence partitioning for process mining with unlabeled event logs. Data Knowl Eng 2011;70(10):821–41. http://dx.doi.org/10.1016/j.datak.2011.05.003.

[12] Bayomie D, Awad A, Ezat E. Correlating unlabeled events from cyclic business processes execution. In: Nurcan S, Soffer P, Bajec M, Eder J, editors. Advanced information systems engineering - 28th international conference, CAISE 2016, Ljubljana, Slovenia, June 13-17, 2016. proceedings. Lecture notes in computer science, vol. 9694, Springer; 2016, p. 274–89. http://dx.doi.org/10.1007/978-3-319-39696-5_17.

[13] Corallo A, Lazoi M, Striani F. Process mining and industrial applications: A systematic literature review. Knowl Process Manag 2020;27(3):225–33.

[14] Bayomie D, Helal IMA, Awad A, Ezat E, Bastawissi AE. Deducing case IDs for unlabeled event logs. In: Reichert M, Reijers HA, editors. Business process management workshops - BPM 2015, 13th international workshops, Innsbruck, Austria, August 31 - September 3, 2015, revised papers. Lecture notes in business information processing, vol. 256, Springer; 2015, p. 242–54. http://dx.doi.org/10.1007/978-3-319-42887-1_20.

[15] Bayomie D, Di Ciccio C, Rosa ML, Mendling J. A probabilistic approach to event-case correlation for process mining. In: Laender AHF, Pernici B, Lim E, de Oliveira JPM, editors. Conceptual modeling - 38th international conference, ER 2019, Salvador, Brazil, November 4-7, 2019, proceedings. Lecture notes in computer science, vol. 11788, Springer; 2019, p. 136–52. http://dx.doi.org/10.1007/978-3-030-33223-5_12.

[16] Sahu M, Nayak GK, Nayak RK. Process model discovery from unlabeled event logs by using non-overlapping sequential distinct event patterns. Int J Eng Res Technol 2020;13:3055–66.

[17] Burattin A, Vigo R. A framework for semi-automated process instance discovery from decorative attributes. In: Proceedings of the IEEE symposium on computational intelligence and data mining, CIDM 2011, part of the IEEE symposium series on computational intelligence 2011, April 11-15, 2011, Paris, France. IEEE; 2011, p. 176–83. http://dx.doi.org/10.1109/CIDM.2011.5949450.

[18] Brzychczy E, Pelech-Pilichowski T, Dworakowski Z. Case ID detection based on time series data - the mining use case. 2024, http://dx.doi.org/10.48550/ARXIV.2410.23846, CoRR abs/2410.23846, arXiv:2410.23846.

[19] Helal IMA, Awad A. Online correlation for unlabeled process events: A flexible CEP-based approach. Inf Syst 2022;108:102031. http://dx.doi.org/10.1016/J.IS.2022.102031.

[20] Bayomie D, Di Ciccio C, Mendling J. Event-case correlation for process mining using probabilistic optimization. Inf Syst 2023;114:102167. http://dx.doi.org/10.1016/J.IS.2023.102167.

[21] van Dongen B. Real-Life Event Logs - Hospital Log. Eindhoven University of Technology; 2011, http://dx.doi.org/10.4121/UUID:D9769F3D-0AB0-4FB8-803B-0D1120FFCF54.

[22] van Dongen B. BPI challenge 2012. 4TU. 2012, Centre for Research Data. Dataset.

[23] Steeman W. BPI challenge 2013, closed problems. 2013, http://dx.doi.org/10.4121/uuid:c2c3b154-ab26-4b31-a0e8-8f2350ddac11.

[24] Steeman W. BPI Challenge 2013, incidents. Ghent University; 2013.

[25] Steeman W. BPI challenge 2013, open problems. 2013, http://dx.doi.org/10.4121/uuid:3537c19d-6c64-4b1d-815d-915ab0e479da.

[26] van Dongen B. BPI challenge 2014: Activity log for incidents. Rabobank Nederland; 2014, http://dx.doi.org/10.4121/uuid:86977bac-f874-49cf-8337-80f26bf5d2ef.

[27] van Dongen B. BPI challenge 2014: Change details. Rabobank Nederland; 2014, http://dx.doi.org/10.4121/uuid:d5ccb355-ca67-480f-8739-289b9b593aaf.

[28] van Dongen B. BPI challenge 2014: Incident details. Rabobank Nederland; 2014, http://dx.doi.org/10.4121/uuid:3cfa2260-f5c5-44be-afe1-b70d35288d6d.

[29] van Dongen B. BPI challenge 2014: Interaction details. Rabobank Nederland; 2014, http://dx.doi.org/10.4121/uuid:3d5ae0ce-198c-4b5c-b0f9-60d3035d07bf.

[30] van Dongen B. BPI Challenge 2015 Municipality 1. Eindhoven University of Technology; 2015, http://dx.doi.org/10.4121/uuid:a0addfda-2044-4541-a450-fdcc9fe16d17.
[31] van Dongen B. BPI Challenge 2015 Municipality 2. Eindhoven University of Technology; 2015, http://dx.doi.org/10.4121/uuid:63a8435a-077d-4ece-97cd-2c76d394d99c.
[32] van Dongen B. BPI Challenge 2015 Municipality 3. Eindhoven University of Technology; 2015, http://dx.doi.org/10.4121/uuid:ed445cdd-27d5-4d77-a1f7-59fe7360cfbe.
[33] van Dongen B. BPI Challenge 2015 Municipality 4. Eindhoven University of Technology; 2015, http://dx.doi.org/10.4121/uuid:b32c6fe5-f212-4286-9774-58dd53511cf8.
[34] van Dongen B. BPI Challenge 2015 Municipality 5. Eindhoven University of Technology; 2015, http://dx.doi.org/10.4121/uuid:679b11cf-47cd-459e-a6de-9ca614e25985.
[35] Dees M, van Dongen B. BPI challenge 2016: Clicks logged in. 2016.
[36] Dees M, van Dongen B. BPI challenge 2016: Clicks NOT logged in. 2016.
[37] Dees M, van Dongen B. BPI challenge 2016: Complaints. 2016.
[38] Dees M, van Dongen B. BPI challenge 2016: Questions. 2016.
[39] Dees M, van Dongen B. BPI challenge 2016: Werkmap messages. 2016.
[40] van Dongen B. BPI challenge 2017. Eindhoven University of Technology; 2017, http://dx.doi.org/10.4121/uuid:5f3067df-f10b-45da-b98b-86ae4c7a310b.
[41] van Dongen B, Borchert F. BPI challenge 2018. Eindhoven University of Technology; 2018, http://dx.doi.org/10.4121/uuid:3301445f-95e8-4ff0-98a4-901f1f204972.
[42] van Dongen B. BPI challenge 2019. 4TU.Centre for Research Data; 2019, http://dx.doi.org/10.4121/uuid:d06aff4b-79f0-45e6-8ec8-e19730c248f1.
[43] van Dongen B. BPI challenge 2020: Domestic declarations. 4TU.Centre for Research Data; 2020, http://dx.doi.org/10.4121/uuid:3f422315-ed9d-4882-891f-e180b5b4feb5.
[44] van Dongen B. BPI challenge 2020: International declarations. 4TU.Centre for Research Data; 2020, http://dx.doi.org/10.4121/uuid:2bbf8f6a-fc50-48eb-aa9e-c4ea5ef7e8c5.
[45] van Dongen B. BPI challenge 2020: Prepaid travel costs. 4TU.Centre for Research Data; 2020, http://dx.doi.org/10.4121/uuid:5d2fe5e1-f91f-4a3b-ad9b-9e4126870165.
[46] van Dongen B. BPI challenge 2020: Request for payment. 4TU.Centre for Research Data; 2020, http://dx.doi.org/10.4121/uuid:895b26fb-6f25-46eb-9e48-0dca26fcd030.
[47] van Dongen B. BPI challenge 2020: Travel permit data. 4TU.Centre for Research Data; 2020, http://dx.doi.org/10.4121/uuid:ea03d361-a7cd-4f5e-83d8-5fbdf0362550.