Process Mining by using Event Logs

Swapnali B. Sonawane
Dept. of Computer Engineering
Dr. D. Y. Patil College of Engineering, Ambi
Pune, India

Ravi P. Patki
Dept. of Computer Engineering
Dr. D. Y. Patil College of Engineering, Ambi
Pune, India

ABSTRACT

Process mining techniques have usual notable attention within the literature for their ability to help within the redesign of complex processes by mechanically discovering models that specify the events registered in some log traces provided as input. Process mining refers to the extraction of process models from event logs. Now real-life processes tend to be less structured and a lot of flexible. Traditional process mining algorithms have issues dealing with such unstructured processes and generate “spaghetti-like" process models that are exhausting to understand. An approach to beat this is often to cluster process instances specified every of the ensuing clusters correspond to coherent sets of process instances which will every be adequately represented by a process model.

To overcome these issues projected system aims to produce associate automatic means for code engineers to get mined models from systematic event logs specification embrace drawback finding, operating to learn others and technical challenge.

This technique at first converts the Systematic Event Logs into some intermediate type like translated tokenized log file and keyword filtered log file. Then this filtered log file format is analyzed to extract the knowledge like Similarity matrix, Frequency count, Most read/write information, database queries and these event logs data measure accustomed build the clusters. Any system would generates the clusters using ActiTraC algorithm to produce refined description of generated models therefore incorrectness and additional overhead in analysis part of model development is removed to extended extent.

This is supported on an repetitious, graded, refinement of the process model, where, at every step, traces sharing similar behavior patterns are clustered along and equipped with a specialized schema. The formula guarantees that every refinement results in an progressively sound model, so attaining a monotonic search.

Keywords

Process mining, event log, process discovery, trace clustering, process model, data mining.

1. INTRODUCTION

We The main objective of process mining encompasses “techniques, tools and methods to discover, monitor, control, data, organizational, and social structures and improve real processes by extracting knowledge from event logs. The basic idea is to extract knowledge from event logs recorded by an information system. Process mining has proven to be a valuable tool for analyzing operational process executions based on event logs. Unfortunately, process mining is most interesting in domains requiring flexibility. An approach using trace clustering[2], i.e., the event log is split into homogeneous subsets and for each subset a process model is created.

In particular, process mining techniques are highly suitable in flexible environments such as Healthcare, Customer Relationship Management (CRM), Product Development, Workflow Management (WFM), Enterprise Resource Planning (ERP), Business to Business (B2B), and Supply Chain Management (SCM) systems and so on [2]. The aim of process mining is the extraction of information about real time processes. The data that is generated during the execution of real time processes in information systems is used for reconstructing process models. Process mining is an innovative approach and builds a bridge between data mining and process management. These models are useful for analyzing and optimizing processes.

Process mining techniques have usual notable attention within the literature for their ability to help within the redesign of complex processes by mechanically discovering models that specify the events registered in some log traces provided as input[7]. Process mining refers to the extraction of process models from event logs. Now real-life processes tend to be less structured and a lot of flexible. Traditional process mining algorithms have issues dealing with such unstructured processes and generate “spaghetti-like"[1] process models that are exhausting to understand. An approach to beat this is often to cluster process instances specified every of the ensuing clusters correspond to coherent sets of process instances which will ever be adequately represented by a process model.

This evaluates identification of various variants for the process is expressly accounted for, supported the cluster of log traces. Indeed, modeling every cluster of comparable executions with a special schema permits us to single out “conformant” models, which, specifically, minimize the quantity of modeled enactments that are extraneous to the process semantics. It judge the goodness of the shaped clusters exploitation established fitness and quality metrics outlined in the context of process mining[9]. The planned approach is in a position to get clusters specified the process models strip-mined from the clustered traces show a high degree of fitness and quality. Further, the proposed feature sets may be simply discovered in linear time creating it amenable to time period analysis of huge data sets.

2. BASIC CONCEPTS

2.1 Process Mining

The first point for process mining is an event log. The event in such a log refers to an activity [15] and is related to a particular process instance. The events belonging to a process instance are ordered. Event logs can store additional data about events. Process mining techniques use supplementary information such as the resource executing or initiating the activity, logs data, event’s time stamp, and data attributes.

Attributes store additional information that can be used for analysis purposes.

An event log contains all recorded events that relate to executed activities in a table. A process model is an abstraction of the real world execution of a process. The event logs as a set of events that are mapped to the same case. The sequence of recorded events in a case is called trace. Process instance model describes the execution of a single process instance. A process model abstracts from the single behavior of process instances and provides a model that reflects the behavior of all instances that belong to the same process. Classifiers ensure the distinctness of cases and events by mapping unique names to each case and event. Cases and events are characterized by classifiers and attributes.

First, analysts use process discovery techniques [18] to evaluate a model from an event log. Analysts then apply conformance checking techniques to diagnose deviations between the event log and initial process model. After that, during model creation, analysts use information from the log to repair or extend the model. They can use time stamps to add timing information such as waiting times and service times to the model. Then the resulting enhanced process model can support decision making.

3. LITERATURE SURVEY

Process mining is an active research field, and includes a number of different techniques:-

The Srikant & Agrawal [4]uses an un-weighted is-a directed acyclic graph hierarchy. A technique that is employed to come up with a dependency graph from an real time event log. During this work they permit the employment of weighted graphs and take away the acyclic condition. The distinction is critical and results not solely in an exceedingly completely different algorithmic program being needed however conjointly in rules possessing a unique linguistics structure.

Dongen [3]introduces Instance graphs, an approach that aims at depicting a graphical illustration of process executions, particularly victimization Event-Driven process Chains (EPCs). The method of process mining techniques generally tries and generates a whole process model from the information non-heritable in an exceedingly single step. Dongen propose a multistep approach. He proposes as, initial models are generated for every individual process instance. Within the final step but, these instance models are united to get an overall model for the complete information set, final step, i.e., aggregating instance graphs. The work is motivated by tools and techniques to come up with instance models like ARIS PPM and In Concert generate instance models which will be taken by their method process mining tool. The results of multi-step approach are often described in several styles of process models.

Medeiros [5]introduces, a technique during which many candidate answers are measure evaluated by a fitness perform that determines however consistent every solution is with the log. Current techniques have issues once mining processes that contain the presence of noise within the logs. Most of the issues happen as a result of several techniques are measure supported native data within the event log. To overcome these issues, Medeiros uses genetic algorithms to mine process models. The most motivation is to profit from the worldwide search performed by this sort of algorithms. The non-trivial constructs are measure tackled by selecting an indoor illustration that supports them. The matter of noise is of course tackled by the genetic formula as a result of, these algorithms are measure strong to noise. the most challenge in a very genetic approach is that the definition of a fitness live as a result of it guides the worldwide search performed by the genetic formula. Genetics miner uses a genetic formula to mine a Petri web illustration of the process model from execution traces. though the formula will mine process models which may contain all the common structural constructs like sequence, choice, similarity, loops, non-free-choice, invisible tasks, and duplicate tasks, and might handle noise.

Ferreira et al [6]introduces the principles of sequence cluster and presents two case studies wherever the technique is employed to get behavioral patterns in event logs. A technique that mechanically teams sequences into completely different clusters so as to spot typical activity patterns. The problem usually encountered in applies is that for processes with a high diversity of behavior solely terribly advanced models may be discovered. Grouping the traces into a lot of solid clusters and discovering separate models for every of them is one strategy to get higher models. Iterative approach supported first-order Markov Chains is employed to bit by bit assign traces to the “best” cluster.

In the initial case study, the goal is to grasp the approach members of a software system team perform their daily work, and therefore the application of sequence cluster reveals a collection of behavioral patterns that are measure associated with a number of the most processes being allotted by that team. Within the second case study, the goal is to research the event history recorded in very technical support information so as to work out whether or not the recorded behavior complies with a predefined issue handling process. During this case, the applying of sequence cluster confirms that each one behavioral patterns share a standard trend that resembles the initial process.

In [1], Joachim Herbs proposed an approach during which the goal is to search out a hidden Markov model (HMM) that represents the structure of the initial real time process. Workflow management systems (WFMS) supply very little action of progress models and their alteration to ever-changing necessities. To support these activities Herbst propose associate approach that induces process models from process instances.

In [2], Wil van der Aalst defines a technique that's able to re-create a Petri-net model from the ordering relations found in an event logs. Alpha algorithmic rule will mechanically extract a Petri net that offers a short model of the behavior seen in a very set of event traces, forward the traces area unit of completed instances and it don’t contain noise. It will with success mine any workflow depicted by a structured workflow net. It will with success discover short loops (of length one and a couple of edges).

In [4], Rakesh Agrawal, Johannes Gehrke, Dimitrios Gunopulos and Prabhakar Raghavan proposed hierarchical clustering algorithmic program that, given an oversized set of execution traces of one process, separates them into clusters and finds the dependency graph on an individual basis for every cluster.

It is an agglomerate (top down) clustering technique. This technique is employed to make a hierarchy of clusters, that shows relations between the individual members and merging clusters of information supported similarity.

In [8], Christian W. Günter proposed trace segmentation techniques address the matter that events are typically logged on a way additional fine-grained level of abstraction. Activity mining bridges the gap between variety of low-level events and a higher-level activity by grouping traces. Supported these activities, less complicated process models will be discovered like world and hierarchical activity mining approach.

In [9], R.P. Jagadeesh Chandra Bose (JC) focused to manage less structured processes in real-life things, abstractions are typically required to get models that may be understood. Here, the manifestations of ordinarily used process model constructs (e.g., loops) are investigated, and also the derived pattern definitions are then used as abstractions for the mining of higher-level activities within the event log.

In [10], Philip Weber defines process mining is actually a machine learning task, however very little work has been done on consistently analyzing algorithms to grasp their basic properties, like what quantity knowledge are required for confidence in mining. This paper proposes a framework for analyzing method mining algorithms. Many processes are viewed as distributions over traces of activities and mining algorithms as learning these distributions. Philip Weber, Behzad Bordbar, and Peter Tiňo use probabilistic automata as a unifying illustration to that different illustration language will be reborn and an analysis of the Alpha algorithmic rule under this framework.

In [11], Can Wang, Xiangjun Dong, Fei Zhou, Longbing Cao propose coupled cluster attributes. It involves the frequency-based intra-coupled similarity at intervals an attribute and therefore the inter-coupled similarity upon value co-occurrences between attributes, additionally as their integration on the object level. Especially, four measures are designed for the inter-coupled similarity to calculate the similarity between 2 categorical values by considering their relationships with alternative attributes in terms of power set, universal set, joint set, and intersection set. The theoretical analysis reveals the equivalent accuracy and superior potency of the measurement on the intersection set, significantly for large-scale data sets. Severe experiments data structure and clustering algorithms incorporating the coupled unsimilarity metric deliver the goods a big performance improvement on state of- the-art measures and algorithms on thirteen UCI data sets that is confirmed by the statistical analysis. Additionally, 2 new coupled categorical clustering algorithms, i.e., CROCK and CLIMBO are projected, and that they each outmatch the first ones in terms of clustering quality on UCI data sets and list information.

In [12], Jianmin Wang, Raymond K. Wong, Jianwei Ding, Qianlong Guo, and Lijie Wen defines selection of process mining algorithms. It is tough to decide on an acceptable process mining algorithm for a given enterprise or application domain. This paper investigates a scalable resolution which will judge, compare, and rank these process mining algorithms expeditiously, and therefore proposes a completely unique framework which will expeditiously choose the process mining algorithms that are most fitted for a given model set. This paper conjointly proposes a metric and technique to pick high-quality reference models to derive an efficient regression model.

4. PROBLEM DEFINITION

To evaluate a new approach using trace clustering, to construct improved process models and to resolve the problem that currently exist in process discovery algorithms such as unable to discover accurate and comprehensible process models out of event logs stemming from highly flexible environments. Also to describe the difference between clusters of process instance that is currently lacking from a cluster evaluation perspective.

5. MOTIVATION

Proposed system findings suggest that software engineers are likely to be motivated according to three related factors such as their need for variety, their implementation strategy, their software architecture & DB Design. Published models in Software Engineering are disparate and do not reflect the complex needs for software engineers in application development stages.

6. SYSTEM METHODOLOGY

This system initially first converts the systematic event logs into some intermediate form such as translated tokenized log file and keyword filtered log file [17]. Then this filtered log file format is analyzed to extract the information and then these event logs information are used to build the clusters. Further system would generates the clusters using ActiTraC[19],[20] algorithm to provide refined description of generated models thus incorrectness and extra overhead in analysis phase of model development is removed to significant extent.

After that these clusters are compared with mined process definition. After this comparison proposed system generates mined process models. System would have to extract systematic event logs from given process information and then transforms the real time event logs to transformed cases(it contains tokenized and filtered logs) and then use the ActiTraC algorithm to constructs clusters . After that these clusters are compared to the standard mined process definition and generate mined process models. Finally using this information it generates a prediction of mined process models. It is defined in fig1.

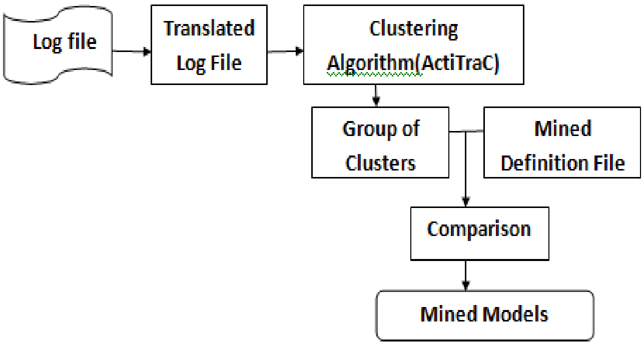

Fig 1: Proposed Architecture Diagram of System

Here take the input from user, which is a systematic event logs file. Convert this event logs file into translated tokenized logs file through the log transformation phase. Then the tokenized logs file is then processed through the detailed filtering process [18] which filters the stop words, common words. On the output of filtering step we apply our rules on selection criteria's of ActiTraC algorithm on the basis of which actors, events, candidate classes, their attributes, their relationships are extracted. Our rules process filtered log file and generates clusters of similarity matrix, mostly used operators/operands, most read/write data, database queries. After that these clusters are compared to the process mining definition and generate mined process models.

7. SYSTEM WORKFLOW

Following diagram shows workflow of system

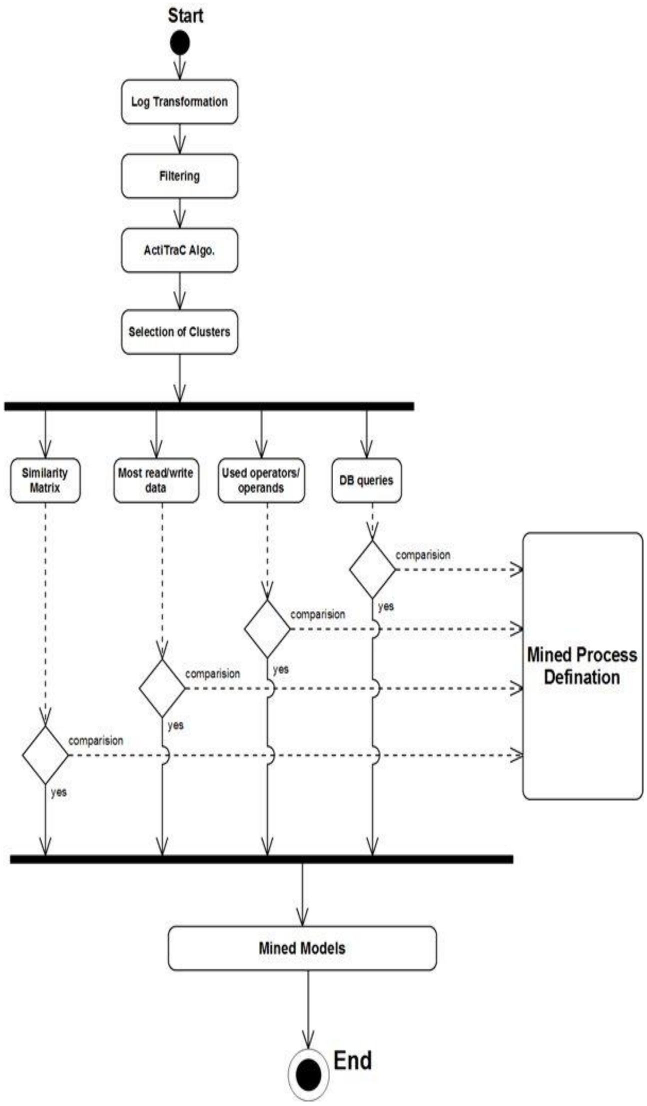

Fig 2: System Workflow Diagram

The proposed system has following steps to generates mined process models from event logs:

7.1 Log Transformation

The input to the proposed system is systematic event log file of real time processes. The event log is a key concept in the field of process mining. An event log consists of a set of traces and a trace is a sequence of events. Every event log, trace or event can contain data attributes. The data attributes of an event contains the resource or timing information. Every type of activity is also called an event class.

An event is a recorded execution of an activity. A trace is a recorded sequence of events that belong to the same case. Events are mapped to cases. Each case has a trace. In an event log, there can be instances where the system is subjected to similar execution patterns or behavior. Discovery of common patterns of invocation of activities in traces can help in improving the discovery of process models and can help out in defining the conceptual relationship between the tasks or activities.

Log transformation is done by several transformation techniques such as String Tokenizer, Raw detailed filtering.

7.1.1 String Tokenizer

A Tokenizer splits a stream of characters (from each individual field value) into number of tokens. There is only one Tokenizer in each Analyzer. The tokenization is the task of separating string into number of pieces, they called tokens, at the same time throwing away some characters, such as punctuation. Also a token is an instance of a sequence of characters in some particular document that are grouped together as a useful semantic unit for processing. Token Filters: Tokens produced by the Tokenizer are passed through a number of token filters that rearrange or remove tokens.

7.1.2 Raw detailed filtering

In this step, the tokenized transformed cases are filtered. It filters the STOP words, COMMON words. A token filter of type stop that removes stop words from token streams. Some extremely common words that would appear to be of little value in helping select documents matching a user need are excluded from the vocabulary entirely. So using a stop list significantly reduces the number of postings that a system has to store. Any group of words can be chosen as the stop words for a given purpose. Some of the most common, short function words, such as a, an, and, are, as, be, for, to, in, the, is, at, which and so on.

7.2 Clustering Algorithm: ActiTraC

This algorithm is used for making clusters from different real life event logs. The proposed system considers the four types of event logs such as

a. Similarity matrix
b. Most Frequently read/write data
c. Most Frequently used operators/operands
d. Database Queries

There are 3 main steps involved in this algorithm -

1. Selection
2. Look Ahead
3. Residual trace Resolution

7.2.1 Selection

The filtered log file is given to the ActiTraC. Then this algorithm performs different actions on data which from event log file. Then number of sub sets of Log file need to be separated, with analyzing repeated steps & grouping them into single unit and is called as cluster.

For e.g. consider log contains {abcdbcdbcdecbd} in this log file {bcd} event is getting executed 4 times so set becomes {bcd, 4}.So the generated subsets are then transferred to the next step i.e. look ahead.

7.2.2 Look Ahead

The output of selection phase is the input of look ahead step.It defines selection criteria in Selection process and makes subsets of event logs, which will act like an input for Residual trace Resolution.

7.2.3 Residual Trace Resolution

This is the final step of algorithm. Residual means "a quantity remaining after other things have been subtracted". It generates number of clusters.

7.3 Comparison of Clusters with Mined Process Definition and Output

This phase performs the comparison of the derived clusters from ActiTraC algorithm with the standard mined process definition. The comparison interprets the mined process models. The mined models which are derived from the comparison phase are predicted for future analysis to find mined process models from event logs.

8. UTILITY

- It is used across all domains over unlimited requirement size of log file.
- It would be very useful in understanding functional requirements .
- It gives list of group of clusters that represents the behavior of system.
- It is used to motivate software engineers in the areas like problem solving, working to benefit others and technical challenge.
- It does not reflect the complex needs for software engineers in application development stages.

9. CONCLUSION

Many Process mining algorithms are unable to discover actions in the process of creation of mined process models from systematic real time event logs of highly flexible environments such as representational bias Inability to represent concurrency, Inability to deal with (arbitrary) loops, Inability to represent silent actions, Inability to represent duplicate actions, Inability to model OR-splits/joins, Inability to represent non-free-choice behavior, Inability to represent hierarchy, Inability to deal with noise, completeness. The key finding is that the published models of motivation in software engineering are disparate and do not reflect the complex needs of software engineers in application development stages. To resolve these inability issues in the development of constructing mined process models from event logs of real time processes, this proposed system use ActiTraC clustering algorithm. To evaluate a new approach using trace clustering in process mining, to evaluate improved process models and to resolve the problem that currently exist in process discovery algorithms such as unable to discover accurate and comprehensible process models out of event logs stemming from highly flexible environments.

10. ACKNOWLEDGMENT

The presented paper would not have been possible without college Dr. D. Y. Patil College of Engineering, Ambi, Pune. I have got support from my family and friends. I thankful to all teachers who guide me, which help me in improving my work, from this I learnt number of new things. So I thankful to everyone.

11. REFERENCES

[1] Joachim Herbst: “An Inductive Approach to the Acquisition and Adaptation of Workflow Models” (1999).

[2] Wil van der aalst, α-algorithm: “Process mining: Overview and opportunities” (2004).

[3] B. F. van Dongen and W. M. P. van der Aalst, Instance graphs : “Multi-phase Process mining: Aggregating Instance Graphs into EPCs and Petri Nets” (2005).

[4] Rakesh Agrawal , Johannes Gehrke , Dimitrios Gunopulos and Prabhakar Raghavan, Hierarchical clustering : “Automatic Subspace Clustering of High Dimensional Data” (2005).

[5] K. A. de Medeiros, A. J. M. M. Weijters, and W. M. P. van der Aalst, Genetic algorithms : “Genetic process mining: An experimental evaluation” (2007).

[6] Ferreira et al, Sequence Clustering: “Techniques for Process Mining Sequence clustering” (2007).

[7] Goedertier et al, Negative events : “Declarative Techniques for Modeling and Mining Business Processes” (2008).

[8] Christian W. Günther : “Activity Mining by Global Trace Segmentation” (2009).

[9] R.P. Jagadeesh Chandra Bose (JC) : “Abstractions in Process Mining: A Taxonomy of Patterns” (2009).

[10] Philip Weber, Behzad Bordbar, and Peter Ti˜no : “A Framework for the Analysis of Process Mining Algorithms” (2013)

[11] Can Wang, Xiangjun Dong, Fei Zhou, Longbing Cao: “Coupled Attribute Similarity Learning on Categorical Data” (2014)

[12] Jianmin Wang, Raymond K. Wong, Jianwei Ding, Qinlong Guo, and Lijie Wen : “Efficient Selection of Process Mining Algorithms” (2013)

[13] Yongkweon Jeon and Sungroh Yoon,“Multi-Threaded Hierarchical Clustering by Parallel Nearest-Neighbor Chaining” (2013)

[14] Wil van der Aalst, Senior Member, “Service Mining: Using Process Mining to Discover, Check, and Improve Service Behavior”(2013)

[15] W.M.P. van der Aalst, Process Mining - Discovery, Conformance and Enhancement of Business Processes. Springer, (2011).

[16] W.M.P. van der Aalst, A.J.M.M. Weijters, and L. Maruster, “Workflow Mining: Discovering Process Models from Event Logs”,(2004).

[17] R.P. Jagadeesh Chandra Bose and W.M.P. van der Aalst, “Context Aware Trace Clustering: Towards Improving Process Mining Results,”(2009).

[18] G. Greco, A. Guzzo, L. Pontieri, and D. Sacca‘, “Discovering Expressive Process Models by Clustering Log Traces,” IEEE Trans. Knowledge and Data Eng., (2006).

[19] M. Song, C.W. Gu nther, and W.M.P. van der Aalst, “Trace Clustering in Process Mining,” (2008).

[20] A.J.M.M. Weijters, W.M.P. van der Aalst, and A.K. Alves de Medeiros,“Process Mining with the Heuristics miner Algorithm,” (2006).