A Multi-Level Process Mining Framework for Correlating and Clustering of Biomedical Activities using Event Logs

[1]Muhammad Rashid Naeem
School of Computer Science and Technology,
Harbin Institute of Technology, Harbin, China
School of Software Engineering
Chongqing University, Chongqing, China

[2]Hamad Naeem, [3]Muhammad Aamir
School of Computer Science and Technology
Sichuan University, Chengdu, China

[4]Waqar Ali
School of Computer Science and Technology,
Nanjing University of Science and Technology
Nanjing, China

[5]Waheed Ahmed Abro
School of Computer Science and Engineering
Southeast University, Nanjing, China

***Abstract*—Cost, time and resources are major factors affecting the quality of hospitals business processes. Bio-medical processes are twisted, unstructured and based on time series making it difficult to do proper process modeling for them. On other hand, Process mining can be used to provide an accurate view of biomedical processes and their execution. Extracting process models from biomedical code sequenced data logs is a big challenge for process mining as it doesn't provide business entities for workflow modeling. This paper explores application of process mining in biomedical domain through real-time case study of hepatitis patients. To generate event logs from big datasets, preprocessing techniques and LOG Generator tool is designed. To reduce complexity of generated process model, a multilevel process mining framework including text similarity clustering algorithm based on Levenshtein Distance is proposed for event logs to eliminate spaghetti processes. Social network models and four distinct types of sub workflow models are evaluated using specific process mining algorithms.**

***Keywords*—biomedical event data; business process modeling; Levenshtein similarity clustering; multilevel process mining; spaghetti process models**

I. Introduction

Information retrieval is a big challenge in IT as the data is rapidly increasing day by day. The growth of data and technology are incredibly high resulting in business process management as a major problem within organizational entities. Today Business processes are more twisted and timely changing compared to old school of thoughts. It is need of any organization to identify, monitor and ensure their business processes are running accordingly to their workflow structure to prevent future losses [1]. Organizations are focusing more on business process improvement to increase concerns and success factors within business [2]. Event driven business process management has become one of emerging trends as it can be applied on businesses processes for compliance monitoring to analyze past business flaws and identify future risks [3]. Thus, process management has become a great importance to organizations and also a need of time in any business environment [4].

Hospitals have greater importance to process management as compared to any other organization as they have problems related to resources, cost and time management. Hospital systems are facing many challenges towards business process management because failure modes are intolerable in hospitals as it can put patients' life at stake. Patients' safety is also a critical factor directly linked to hospital business processes. In past, there are many incidents in medical history due to surgical mistakes or wrong treatment taken on patients possibly due to poor management or work pressure over resources. An abstract view of hospital system can be described using an enterprise architecture. Ahsan et al describes importance of healthcare enterprise architecture as it has more potential to facilitate healthcare units and business processes as a strategy to reduce critical factors and improve business processes [5]. Financial problems are also becoming major concerns in hospital systems. Freund describes a survey conducted by American College of Healthcare Executives and reviews taken from 338 executives of hospitals about hospital business concerns. Report highlighted financial concerns as one of the top concern in hospital managements which could result in resource and technology management problems [6]. A proper resource management can play a significant role to improve hospital services quality. Technology can also play important role to visualize hospital problems. Using latest ERP tools, fraction of business process can be utilized which are helpful in making future business decisions.

Biomedical processes are one of the most ignored parts in hospital business process management as they are mostly based on blood examinations taken on patient subjects. Biomedical processes are unstructured, twisted and based on time series, making it impossible for business analysts to do process modeling for them due lack of internal knowledge and code sequences. Another problem in biomedical process management is that there are thousands of biomedical experiments taken on patients on regular basis. Therefore process modeling for them can only be possible through an automatic process generation system.

II. Background and Motivation

Process mining is a new emerging trend in business process management. It provides different ways to extract business processes using knowledge management techniques without need of any background understanding of subject organization. For further discussion, this section is divided into two parts. At First, we provide a basic overview of process mining and its applications to business process management domain. Secondly, we evaluate business management problems in healthcare and importance of process mining to solve these problems.

A. Process Mining Overview

Process mining is the experimental, interdisciplinary scientific domain that provides popular algorithms to plot process models from event logs. Event logs consist of events which can be extracted from historical data i.e. ERP systems, distributed databases or SAP systems etc. [7]. As data mining provides prediction analysis from big data, likewise process mining prediction provides business process analysis from big event logs. Nowadays, Big data has become an essential ingredient of business process management to improve organization business entities but it has itself complexity issues. Therefore, enhanced business process management techniques such as process mining are required to provide proper validation and verification of business entities. Currently, three different types of process mining is being used in IT industry as shown in figure 1.

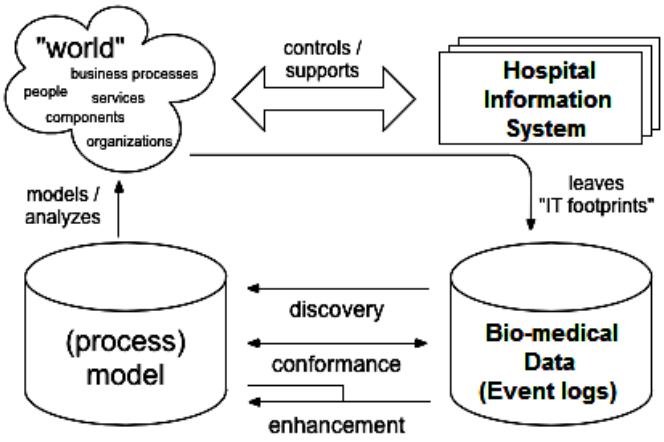

Fig. 1. Process mining as Discovery, Conformance and Enhancement

The diagram shows interaction of real world with healthcare information system. The real world always leaves footprints in IT systems' databases. For example, an employee has done "register" task which has been saved in database of IT system could be a footprint of that employee to "register" process for case "patient ID". We can further analyze these footprints to model the real world behavior using event logs as discovery, conformance and enhancement. In discovery, we generate process model from event log while conformance checking algorithms are used to replay that event log over generated process model to analyze compliance of process model with event log. Lastly, Enhancement algorithms are applied to enhance previously generated process models with respect to event log.

B. Process Mining Significance for Biomedical Processes

Biomedical data can also be referred as big data due to machine generated codes while stored data is consisting of textual information without any standardized terminology. There are difficulties and challenges of understanding and extracting useful information from biomedical data. Structure complexity issues also imposed limitations on biomedical data [8]. Extracting biomedical processes from biomedical data can only be possible using automatic process generation techniques i.e. process mining and process prediction etc. Many useful methods are discovered to analyze biomedical processes such as: McNames proposes use of biomedical filter to estimate event rates and extract point processes within in biomedical signals [9]. Augen proposes using of bioinformatics and information technology together has possibility to do process discovery in drugs [10]. Bose and Aalst describes process mining techniques can be used to extract non-trivial process related knowledge and analyze interesting insights to biomedical data which can later be used for performance analysis and other mathematical operations [11]. Petri net diagrams (also known as place/transition net) provide a graphical view to analyze business processes which are usually generated using process mining algorithms. Process mining i.e. discovery, conformance and enhancement could provide automatic analysis to biomedical process. Chaouiya describes petri nets emerged as promising tool to analyze biological networks efficiently [12]. Ferreira et al propose sequence clustering technique for bioinformatics to extract sequence behaviors using process mining [13]. Xing et al also propose an algorithmic approach to mine distributed bioinformatics workflows which can be applied within hospital systems to handle concurrence and recurrence of restricted bioinformatics workflow processes [14].

Hospital business processes are critical and changing timely, therefore process improvement is becoming requirement of time and one of main concerns of hospitals especially for healthcare technologies and informatics [15][16]. From previous observations, it can be concluded using process mining techniques, it is possible to make proper business process models from hospital data logs to visualize concerns especially in biomedical domain. On other hand, extracting biomedical processes is also a critical requirement of this century. Due to increase in biomedical equipment usage, biomedical record is rapidly converting into big data making it difficult to do estimations and performance analysis at business process level. Extracting useful information is now possible using number of open source and commercial information processing software's available but most of information gained is related to analytical estimations only. Extracting business processes from bioinformatics is a big challenge in biomedical domain which is main contribution of this paper.

III. Preprocessing of Data using *"LOG Generator"* Tool

In this section, we provide an overview of case study and techniques to convert non-event biomedical data into events by querying among multiple dataset tables. Secondly, we provide

"LOG Generator" tool algorithm which is used generate event log for from preprocessed event data.

A. Case study Overview

For biomedical case study, Hepatitis Patients' data has been selected for process mining which is taken from ECML/PKDD discovery challenge website [17]. The data set contains examinations of Hepatitis B and C on patients admitted to academic hospital. The dataset contains huge amount of time series experimental data. All experiments are taken from different laboratories and medical facilities between year 1978 and 2001.

Dataset consists of seven tables. First table "pt_ e030704" contains information related to patient's identity i.e. IDs, birth dates etc. "bio_e030704" table contains information about biopsy of patients. "ifn_e030704" table contains interferon therapy information performed on patients. "hemat_e030704" contains information about hematology experiments while "ilab_e030704" and "olab_ e030704" tables contain different experiments taken inside and outside hospital laboratories. Lastly, "labn_ e030704" table contains measurement units for in-hospital laboratories.

In-hospital data contains results of 230 distinct examinations while out-hospital data consists of 753 distinct examinations performed on 771 hepatitis patients admitted to academic hospital.

B. Preprocess Event Data

In preprocessing, we extract real time events from non-event data sources i.e. data table as we discussed in section 3.1 case study overview. There are four required elements of an event needed to be extracted to create a business event are "Case ID", "Activity", "Timestamp" and "Resource". Almost every distributed information system has fractions of events i.e. hospital patient's database, transaction log etc. Let's take example of hepatitis patients' data to illustrate an events extraction. A subject patient is a "Case ID", time is used as "Timestamp" and experiment name is an "Activity" while facility can be used as a "Resource" for that activity. There are five tables in hepatitis patients database from which events can be extracted by querying with patient info table. For this paper, we are using biopsy table to show events extraction practice from biomedical data logs. In figure 2, an event extraction technique for biopsy processes is presented.

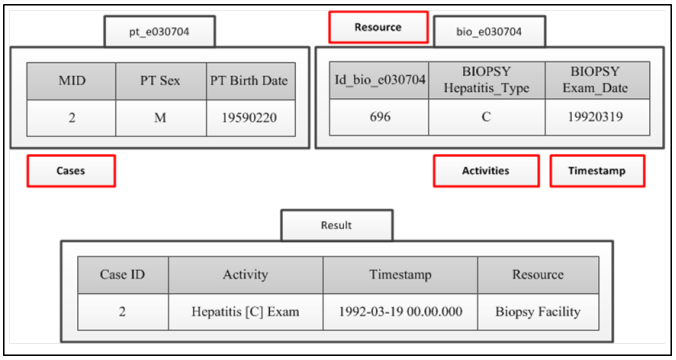

Fig. 2. A resultant event extracted by querying patient info and biopsy table

"MID" field of patient info table is treated as "Case ID" from "pt_e030704" table while "Biopsy Hepatitis_ Type" is used as "Activity" and "Biopsy Exam_ Date" is used as timestamp from "bio_e030704" table. There is no specific resource field provided in biopsy table, therefore "Biopsy Facility" is used as a resource for respective event. The Date is converted to timestamp format while "C" is name of blood experiment which become "Hepatitis [C] Exam" process. All remaining events are extracted from other tables using similar extraction technique.

After extraction, all event data is exported to sing *CSV* file collectively having more than 1.6 million events with 727 cases and 1067 distinct activities. The smallest case consists of 3 events while the longest case has 13257 events.

C. Event Log Generator Tool

Hepatitis patient's event data is huge in size and needed to be converted into XES (eXtensible Event Stream) event log format to apply process mining. XES is successor of old MXML (Mining eXtensible Markup Language) event log. Contrary to MXML, XES is simple without any restrictions of classifying data attributes. XES user manual and latest XES standard definitions are presented in [18]. There are also many XES event log generator tools available today and widely used for process mining. The most popular and open source XES event log generator tools are "openXES" and "XESame". There are some complexities of usage issues in these log generator tools i.e. java jdbc connections and query setting are required to build a successful event log which is too difficult for newbies and required deep understanding of tools to use them properly. The user manual guide for "penXES" and "XESame" log generator tools are presented in [19] and [20] respectively. Due to large size of data, generating an event log is not an easy task as it causes computer memory and application heap size problems in tools as well querying takes lot of time for generating event log which are not handled properly in mentioned tools. To tackle such problems a "LOG Generator" tool has been developed which uses same data setting of CSV event data file as described in 3.3 preprocess event data section. Our tool uses an openCSV api to read biomedical event data ensuring faster event log generation as it is designed to handlle large CSV datasets for read and write purposes. To elaborate internal structure of "LOG Generator" tool a pseudo code of EventBuilder algorithm is presented in 3.3.1.

"LOG Generator" is a user friendly tool and doesn't require any prior usability knowledge. *PrintWriter* and *PrintReader* utility classes are used for reading and writing data in XES event log format. Final output of tool is *Eventlog.xes* file which is an XES format event log file have compatibility to any popular process mining tool to apply process mining algorithms. The time format of biomedical data is usally in *YYYYMMDD* format which is not a standard in any of widely used programming languages. For this purpose a *TIMESTAMP-CONVERTER* procedure is designed to convert date into *yyyy-mm-ddT hh:mm:ss.sss+GMT* timestamp format which is used as a standard in Process mining. The meta structure of event log generated using "LOG Generator" tool is shown in table 1. The average time tool takes to generate 1.6 million events log is 24 secs.

Algorithm 3.3.1 : EVENTBUILDER (csv)

procedure $TIMESTAMPCONVERTER\,(n,\ D)$
$f \leftarrow DATEFORMATER\,(`yyyy - mm - dd'\mathrm{T}'hh : mm : ss.sss + GMT')$
$t \leftarrow \{\}$
if $n \geq 8$
then $\begin{cases} y \leftarrow \{D_0,\ ...,\ D_3\} \\ m \leftarrow \{D_4,\ D_5\} \\ d \leftarrow \{D_6,\ D_7\} \\ t \leftarrow CONCAT(y, '-', m, '-', d) \end{cases}$
$timestamp = SETFORMAT(f, t)$
return $(timestamp)$
main
$DT \leftarrow READALL(csv)$
$SORT(DT)$
$size \leftarrow LENGTH(DT)$
$output \leftarrow CONCAT(output, '<LOG>')$
$x \leftarrow 0$
while $x \leq size$
do $\begin{cases} e \leftarrow DT[x] \\ \textbf{if } x = 0 \\ \quad \textbf{then } \{\ output \leftarrow CONCAT(ouput, '<trace>') \\ e_{time} \leftarrow TIMESTAMPCONVERTER\,(LENGTH\,(e_{time}), e_{time}) \\ event \leftarrow CONCAT('<event>', '<e_{caseID}>', '<e_{activity}>', '<e_{time}>', '<e_{resource}>', '</event>') \\ \textbf{if } x + 1 > x \textbf{ and } x \neq \{0 \textbf{ or } size\} \\ \quad \textbf{then } \begin{cases} output \leftarrow CONCAT(output, event) \\ output \leftarrow CONCAT(output, '</trace><trace>') \end{cases} \\ \textbf{else} \\ \quad \textbf{then } \{\ output \leftarrow CONCAT(output, event) \\ \textbf{if } x = size \\ \quad \textbf{then } \{\ output \leftarrow CONCAT(ouput, '</trace>') \\ x \leftarrow x + 1 \end{cases}$
$output \leftarrow CONCAT(output, '</LOG>')$
$WRITEFILE(ouput\ as\ \textbf{'EventLog.xes'})$

IV. Complexity Analysis and Proposed Framework

For complexity analysis and process model generation, event log is imported to Prom6 open source tool. We use heuristic miner algorithm to generate process model as it works better with larger size event logs. Heuristic miner uses frequencies and sequences of events by ignoring infrequent paths. It uses casual dependencies and AND/XOR split joins to construct process models [21]. To mine dependencies using heuristic miner, following equation is presented.

$$a \Rightarrow_W b = \left(\frac{|a >_W b| - |b >_W a|}{|a >_W b| + |b >_W b| + 1} \right) \tag{1}$$

Where $a \Rightarrow_W b \in \{-1, 1\}$ and $a, b \in T$ while "W" is an event log over trace "T"

For all non-observable tasks, depending relation equation is used in heuristic miner as follows:

TABLE I. Meta Format of XES Event Log Generated using LOG Generator Tool

```
<log xes.version= "1.0" xmlns= "http://www.xes-standard.
org" xes.creator= "Log Generator Tool">
<!– Extensions –>
<global scope= "trace">
<string key= "concept:name" value= "name"/>
</global>
<global scope= "event">
<string key= "concept:name" value= "name"/>
<string key= "org:resource" value= "resource"/>
<string key= "lifecycle:transition" value= "transition"/>
<date key= "time:timestamp" value= "2001-04-14T05:40:
17.017+8:00"/>
<string key= "Activity" value= "string"/>
<string key= "Resource" value= "string"/>
</global>
<trace>
<string key= "concept:name" value= "1"/>
<string key= "creator" value= "LOG Generator Tool"/>
<!– Events –>
<event>
<string key= "concept:name" value= "Hepatitis [C] Exam"
/>
<string key= "org:resource" value= "Biopsy Facility"/>
</event>
<!– Traces –>
</trace>
</log>
```

$$a \Rightarrow_W b \wedge c = \left(\frac{|b >_W c| - |c >_W b|}{|a >_W b| + |a >_W c| + 1} \right) \tag{2}$$

Where $a \Rightarrow_W b \in \{-1, 1\}$ and $a, b, c \in T$, "W" is an event log over trace "T" while "b" & "c" are in depending relationship with "a"

Correctness/Fitness of generated model is measured using Continuous Parsing Measure (CPM) and equation of CPM is given as:

$$CPM = \frac{1}{2}\frac{(e - m)}{e} + \frac{1}{2}\frac{(e - r)}{e} \tag{3}$$

Where e = events, m = total no. of missing activated inputs and r = no. of remaining activated outputs

One of the main advantages of heuristic miner is it deals with noise and exceptions efficiently and focuses on main process workflow instead of mapping every possible path resulting in reduced spaghetti processes. Testing conformance of event log with process model i.e. replaying all events over process model is also memory and time consuming while heuristic miner uses dependency alignments for each dependency to calculate fitness of generated model. Process model generated using heuristic miner having fitness of 0.78 is shown in figure 3.

Fig. 3. Spaghetti process model generated using heuristic miner

The processes and nodes in figure 3 are twisted and too many giving it a spaghetti look. Therefore current process model can't be used as business process management solution. To resolve this issue, a clustering strategy based multilevel processing framework for event logs is proposed to eliminate these spaghetti processes.

A. Multi-Level Process Mining Framework

To eliminate spaghetti processes and based on structure of biomedical activities, we propose multi-level process mining framework for event logs to generate multi process models shown in figure 4. Clustering of similar and less-frequent processes with removal of non-frequent activates is applied to reduce complexity of main workflow process model. Workflow models are further divided into multi sub models based on their complexities. The main workflow process model will be "Level 0" in framework while corresponding models are "1" ,"2" and so on.

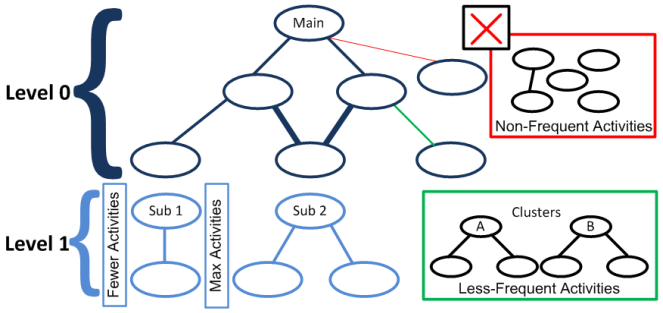

Fig. 4. A multi-level process mining framework to correlate & cluster event logs

Sub process models are further classified into four distinct groups of categories based on frequency of activities. First type of process model consists of fewer activities which mean resulting model will be sound. Second type of process model has extreme number of activities where total number of activities is also huge. The complexity of such process models is same as of main workflow model therefore they are further simplified using "Level 0" techniques. Other types of activities are those having fewer activities as well as non-frequent activities. We further elaborate different properties of framework in next sub sections.

1) Removal of non-frequent activities: The non-frequent activities will be removed from event log as they cant be used as a business activity. For example in hepatitis patients' event log there are more than 100 activities whose occurrence is less than 0.00006% of the total event log. In general, they are automatically ignored by process mining algorithms when putted to noise threshold but it will also affect accuracy and fitness of process models. Therefore to prevent this, all those activities having frequency of occurrence less than equal to 100 are removed from event log.

2) Grouping of Less-Frequent Activities: Second strategy in framework is to reduce spaghetti processes by grouping of those activities having lesser frequency of occurrence in event log as one. For instance, we have created two groups named as "Cluster A" and "Cluster B" where "Cluster A" consists of activities whose frequency ranges between 101 to 1000 while "Cluster B" consists of activities whose frequency ranges between 1001 and 10000 collectively covering 17000 and 102000 events respectively.

3) Similar-Frequent Word Clustering Algorithm: To cluster set of similar and frequent activities within event log, we propose word similarity clustering algorithm for frequent activities. Firstly, all distinct activities in event log are extracted in separate CSV file. Then we use three strategical measures in our algorithm as follows: To extract all possible words within file using strings division, a "Term Frequency" formula is presented which calculate ratio of each term occurrence within CSV file.

$$tf(t,d) = \frac{f_d(t)}{\max\limits_{\omega \in d} f_d(\omega)} \tag{4}$$

Where "d" = document i.e. CSV file and $f_d(t)$ = frequency of term "t" in document "d"

To check similarity among extracted words, Levenshtein Distance wording matching technique is applied through cross multiplication matching between distinct activities in CSV file and terms extracted form sub division of CSV data fields. The documentation for Levenshtien Distance is presented in [22] and formula for calculation of Levenshtien Distance is given as follows:

$$lev_{a,b}(i,j) = \begin{cases} \max(i,j) & if\ \min(i,j) = 0 \\ \min \begin{cases} lev_{a,b}(i-1,j)+1 \\ lev_{a,b}(i,j-1)+1 \\ lev_{a,b}(i-1,j-1)+1_{(a_i \neq b_j)} \end{cases} & else \end{cases} \tag{5}$$

Where "a" and "b" are two strings passed into the loops for similarity distance matching

After calculating distance of all elements in loops, clusters will be made of those having less distance among matching string while duplicate values will be removed by using sorting and distinct value selection techniques as given below:

$$f_{(a,b)}(x) = \left\{ \begin{array}{l} SORT(x \Rightarrow x[a]) \\ DISTINCT(x \Rightarrow x[a]) \end{array} \right\} \tag{6}$$

Where "a" and "b" are cells in multi valued array "f" while sorting and distinct selection is made using "a" column of array.

The detailed pseudo code for clustering and correlating events for event log is described in algorithm 4.4.1.

Algorithm 4.1.1 : FREQCLUSTERBUILDER (csv)

procedure $TERMEXTRACTOR\,(t)$
$t \leftarrow VALIDATEREGEXP\,(t, `[\hat{}A-Za-z0-9]')$
$tf_{1,...,n} \leftarrow SPLITER(t)$
return (tf)
main
$a \leftarrow READALL(csv)$ **where** $csv = \{a_1, a_2, a_3, ..., a_n\}$
$size \leftarrow LENGTH(DT)$
$b \leftarrow \{\}$
$x, y \leftarrow 0$
while $x \leq size$

do
- $t \leftarrow TERMEXTRACTOR\,(DT[x])$
- **while** $y \leq LENGTH(t)$
  - **do**
    - $ADD(a,\ t[y])$
    - $y \leftarrow y + 1$
- $x \leftarrow x + 1$

$a = DISTINCT(a)$ **(i)**
$b = DISTINCT(b)$ **(ii)**

Comment: Now applying cross multiplication among *"a"* and *"b"* arrays where a = {all distinct extracted terms} & b = {all distinct extracted sub terms}

$x, y \leftarrow 0$
$output \leftarrow \{\}$
while $x \leq LENGTH(a)$

do
- **while** $y \leq LENGTH(b)$
  - **do**
    - **if** $b[y].FINDIN(a[x])$
      - **then**
        - $d = \begin{cases} \text{LEVDISTANCE}(a[x], \\ LENGTH(a[x],), \\ b[y],\ LENGTH(b[y])) \end{cases}$
        - $ADD(output, \{a[x],\ b[y],\ d\})$
    - $y \leftarrow y + 1$
- $x \leftarrow x + 1$

Comment: Now sorting output by *"a"* then *"d"* and selecting first row based on *"a"* from *"output"* array.

$SORT(output,\ a \Leftarrow a[0],\ a[2])$
$output \leftarrow DISTINCT(output,\ a \Leftarrow FIRST(a))$
$WRITEFILE(output\ as\ \textbf{csv}_{\textbf{(a,b,d)}})$

After executing algorithm 4.1.1, a CSV file is returned containing three columns "a", "b" & "d", where "a" contains biomedical activities while "b" contains clusters of these biomedical activities in "a". "c" contains distance measured using Lev. distance methodology. Clusters are applied to 1.6 million events using database queries and sample of resultant event data is shown in table 2.

TABLE II. RESULT OF CLUSTER ALIGNMENTS OF EVENTS BY USING DATABASE QUERYING

Case ID	Activity	Cluster	Timestamp	Resource
72	HCV-AB	HCV	1993-08-09T...	ILab Facility
90	B-LIP	Cluster B	1994-06-01T...	ILab Facility
100	ZTT	ZTT	1994-08-24T...	OLab Facility
701	Hepatitis C	Cluster A	1995-08-01T...	Biopsy Facility
752	U-K	U-	1994-10-24T...	ILAB Facility
757	DNA-II	DNA	1997-04-07T...	OLab Facility
.....n	.....n	.....n	.....n	.....n

To visualize graphical view of events' variations due to framework approaches, *x-axis* and *y-axis* are randomly assigned to nearby cluster points by treating each cluster as centroid on rapid miner scatter graph visualizer. Non-frequent, less-frequent and clusters are visualized separately in figure 5 a, b and c.

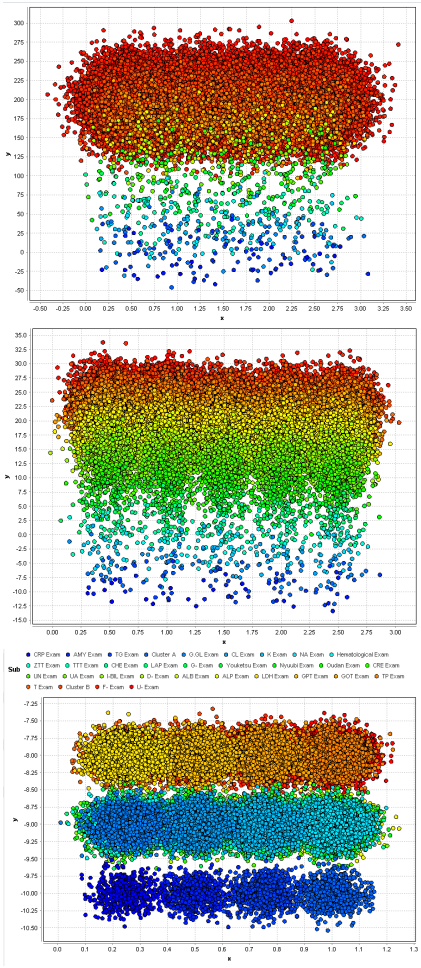

Fig. 5. **(a)**Initial: Event data having both non-frequent and less-frequent activities **(b)** Event data after removal of non-frequent and grouping of less-frequent activities **(c)** Event data after applying similar-frequent clustering algorithm having fewer groups of activities

V. CASE STUDY ANALYSIS

There are five different types of workflow process models generated using multi model process mining framework approaches. At "Level 0", main workflow model is derived while at "Level 1", four distinct types of sub workflow models are generated. They are briefly discussed in coming sections.

A. Main workflow process model

To generate workflow process model, we use same heuristic miner algorithm as described in section 4 complexity analysis part. The generated workflow model after using framework approaches is shown in figure 6.

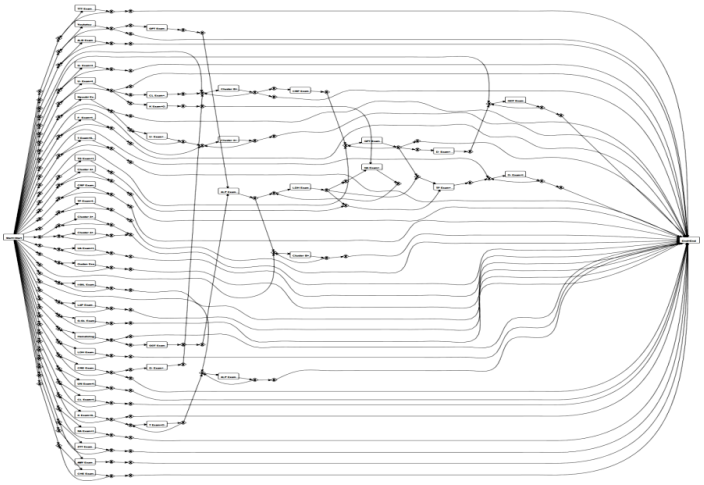

Fig. 6. Main BPMN workflow model generated using heuristic miner through framework approaches

Difference between two diagrams is clearly seen in both figure 3 and 6. Before applying process mining framework, figure 3 illustrates spaghetti and cohesive processes. Conversely, Figure 6 is generated through framework approaches on event log uttering clear workflow nodes with visible spits and joins making it an effective model for biomedical BPM solution. Fitness is another major factor associated with usability and trustworthiness of process models. The fitness of this model is also increased to 0.954 compared to process model in figure 3 which is 0.78 on a scale of 1.

B. Sub workflow process models

Sub workflow process models consist of pertinent spaghetti processes eliminated from main workflow process model and fall in "Level 1" of multi-level process mining framework. Based on hepatitis patients data, several sub workflow models can be derived from event logs where some of them have identical criticality as of main workflow process model. Some of them are comprising of fewer activities and doesnt required any additional process modeling. Based on nature of event logs, workflow models are further divided into four distinct groups as follows:

1) events with fewer event classes workflow model: The first group of event logs has small number of events with fewer event classes. We take "DNA Exam" event log for instance. Event log consists of 206 cases. The timestamp falls between years 1984 to 2000. Event log have 1161 events with 18 event classes and one originator (resource). The process model shown in figure 7 is derived using "Mine Petri Net using Visual Inductive Miner" algorithm provided in Prom6 tool. Visual inductive miner ensures number of input token within petri net are equal to no of output tokens hence ensuring maximum fitness [23]. Noise threshold is added to 0 to map all possible transitions resulting in fitness of derived model to 1.

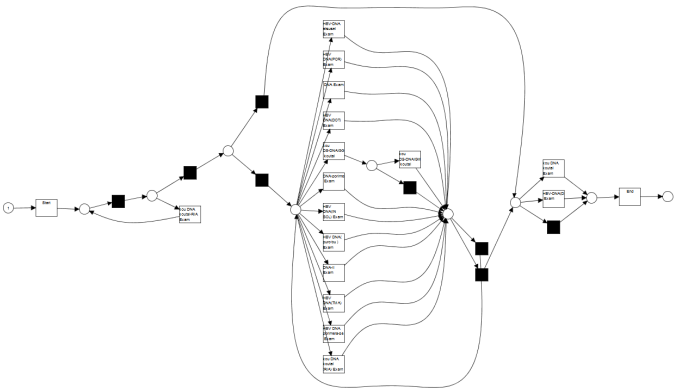

Fig. 7. Petri net sub workflow process model derived from DNA event log using inductive miner

2) Fewer events with extreme event classes workflow model: The second group of event logs has less number of events with extreme number of distant event classes. "Cluster A" event log have fewer events and extreme event classes as it consists of business activities having event frequency rate less than 1000. We apply same "Mine Petri Net using Visual Inductive Miner" algorithm in Prom6 and add noise threshold to 60% to avoid spaghetti processes. Due to 60% noise threshold, the generated model has too many deviations and fitness of derived model is not as much to be used as business process management solution.

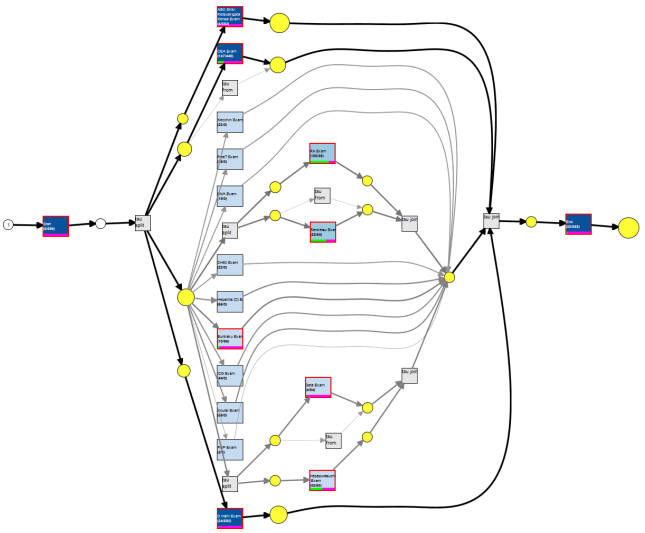

Fig. 8. Performance analysis of petri net sub workflow process model derived from "Cluster A" event log using "Align Log and Model Repair Model for Repair (global Cost)" algorithm

To estimate fitness and fitness cost for each missing token in petri net workflow process model, we use "Align Log and Model Repair for Repair (global Cost" algorithm for performance analysis as shown in figure 8. The repaired workflow model has conformance for only 714 traces while log fitness is 0.57. The calculated global repair cost for the model is 1.74 which is based on number of nonaligned tokens.

Using global cost repair algorithm, It is also possible to trace individual case or individual activity to provide operational support towards trace completion using petri nets.

3) Extreme events with fewer event classes workflow model: Third group of event logs consist of extreme number of events with fewer event classes. For example "F-Exam" event log has 150932 events and 8 distinct event classes. As there are few event classes, therefore generated model will be simple and clear. Noise threshold is also unnecessary due to less number of activities. Therefore, resulting model fitness will be 1 as it covers all paths and activities with in event log similar to figure 7.

4) Extreme events with extreme event classes workflow model: The fourth and last group of event logs consist of extreme number of events with extreme distinct event classes. As the complexity of such event logs is same as of main event log for which multi-level process mining framework is proposed. Therefore it can be further customized using any of three complexity analysis techniques proposed in section 4. Fuzzy miner algorithm can also be useful for such event logs as it makes mini clusters within generated fuzzy model to enhance model detailed view [24]. For instance, we use "U- Exam" event log covering 704 cases 33 event classes and 164095 events. It is also the largest cluster in hepatitis patient's event log. "Mine for fuzzy model" algorithm is used by applying fuzzy inputs to balance fitness and detailed view. Generated fuzzy model is shown in figure 9 a with detailed view of one cluster in figure 9 b. "Model Detail" is 92.68% and "Model Conformance" is 87.26%. Based on proposed framework the cluster falls in "Level 2" of multi-level process mining framework.

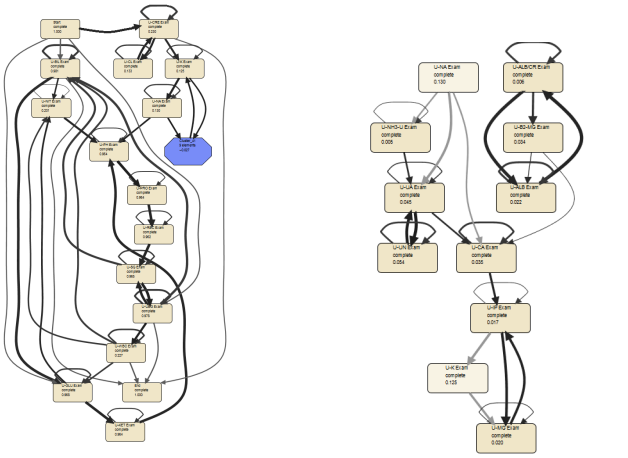

Fig. 9. **(a)** Fuzzy workflow process model generated from "U-" Event log **(b)** Detailed view of fuzzy cluster (blue color business activity) shown in figure 15(a)

C. Social Network Analysis

Organizational perspective plays an important role in applying proper business process management within any organization. Social network models can be used to visualize social behavior and work distribution among organizational resources. Using social network algorithms provided in process mining tools, it is possible to visualize social behavior. Aalst et al terms mining social networks as a virtuous and cost-effective business process analysis technique if combined workflow model concepts with social network analysis to build social network based on hand over work from one performer to another [25].

In biomedical business process environments social networks can be useful to analyze workloads and interaction within hospital resources. As hospital resources strongly dependent on each other i.e. a biopsy cannot be performed without biomedical experimental results, therefore a proper resource management is required to make any decision on resources workload. To visualize organizational perspective, a social network based workflow model is generated using DISCO process mining tool and presented in figure 10.

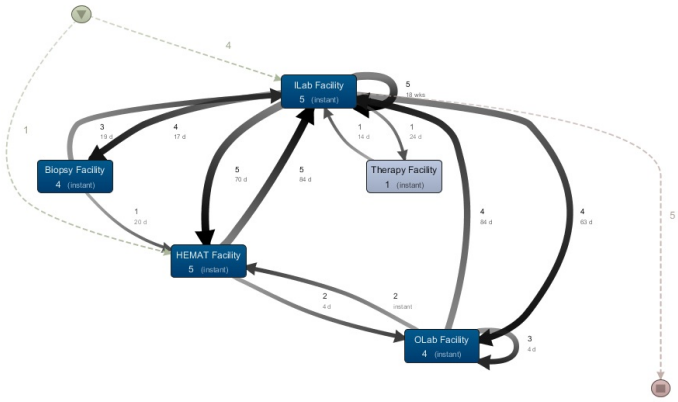

Fig. 10. Visualization of resources executing business activities and their responsibilities using social network mining

VI. Conclusion

Current research focused on implication of process mining in biomedical domain. For case study, an unstructured and time series hepatitis patients' data of academic hospital is utilized. Preprocessing techniques are proposed for events extraction and event logs are generated using "LOG Generator" tool designed to handle huge datasets. To reduce complexity and spaghetti processes in workflow process model, a multi-level process mining framework is envisioned with complexity reduction techniques and a clustering algorithm. Through framework, distant activities are clustered while less-frequent are removed from event logs. The resultant model has shown comprehensible detailed view with greater fitness. Besides, four distinct groups of event logs are elaborated using process mining algorithms to generate sub process models with optimal soundness. Social network model is presented to illustrate organizational behavior and resources. The methods and techniques proposed in research work are helpful for scientific community to apply automatic process modeling from complex and huge bio-medical datasets. In future work, we will extend and apply these methods to solve process modeling problems for other complex and similar series domains.

References

[1] W. X. Mu, F. Benaben, and H. Pingaud, "A Methodology Proposal for Collaborative Business Process Elaboration Using a Model-Driven Approach", *Enterprise Information Systems*, 2015, Vol. 9 No. 4, pp. 349-383

[2] Y. Liu and R. Aron, "Organizational Control, Incentive Contracts, and Knowledge Transfer in Offshore Business Process Outsourcing", *Information Systems Research*, 2015, Vol. 26 No. 1, pp. 81-99

[3] F. Koetter and M. Kochanowski, "A Model-Driven Approach for Event-Based Business Process Monitoring", *Information Systems and E-Business Management*, 2015, Vol. 13 No. 1, pp. 5-36

[4] V. Rajarathinam, S. Chellappa, and A. Nagarajan, "Conceptual Framework for The Mapping of Management Process with Information Technology in a Business Process", *Scientific World Journal*, 2015, No. l983832

[5] K. Ahsan, H. Shah, and P. Kingston, "Patients' Processes in Healthcare: An Abstract View through Enterprise Architecture", *In: Proc. Int. Conf. on Information Management and Evaluation*, Cape Town 2010, pp.459-466

[6] L.M. Freund, "American College of Healthcare Executives Announces Top Issues Confronting Hospitals: 2013", retrieved on 07/03/2015 from http://www.ache.org/pubs/Releases/2014/top-issues-confronting-hospitals-2013.cfm

[7] W.M.P van der Aalst, "Process Mining: Discovery, Conformance and Enhancement of Business Process", *Springer Verlag*, Eindhoven 2011

[8] R.R. Brinkman, M. Courtot, and D. Derom et al, "Modeling Biomedical Experimental Processes with OBI", *In Proc. Of the Bio-Ontologies: Knowledge in Biology*, Stockholm 2009, Vol. 1 Supp. 1, pp. 1-11

[9] J. McNames, "Optimal Rate Filters for Biomedical Point Processes", *In Conf. Proc. of Engineering in Medicine and Biology Society EMBS*, Shanghai 2006, pp. 145-148

[10] N.S. Buchan, D.K. Rajpal, Y. Webster, and C. Alatorre, "The role of translational bioinformatics in drug discovery", *Drug Discovery Today*, 2011, Vol. 16 No. 9, pp. 426-434

[11] R.P.J.C. Bose and W.M.P. van der Aalst, "When Process Mining Meets Bioinformatics; IS Olympics", *Information Systems in a Diverse World*, 2012, Vol. 107, 202-217

[12] C. Chaouiya, "Petri Net Modeling of Biological Networks", *Briefings in Bioinformatics*, 2007, Vol. 8 No. 4, 210-219

[13] D. Ferreira, M. Zacarias, M. Malheiros, and P. Ferreira, "Approaching Process Mining with Sequence Clustering: Experiments and Findings", *Lecture Notes in Computer Science*, 2007, Vol. 4714, 360-374

[14] J. Xing, Z. Li, Y. Cheng, and F. Yin, "Mining Process Models from Event Logs in Distributed Bioinformatics Workflows, *In Proc. of the 1st Int. Symposium on Data, Privacy and E-Commerce*, Chengdu 2007, pp. 8-12

[15] M.H. Yarmohammadian, H. Ebrahimipour, and F. Doosty, "Improvement of Hospital Processes through BPM in Qaem Teaching Hospital: A Work in Progress", *Journal of Education and Health Promotion*, 2014, Vol. 3 No. 111, pp. 1-10

[16] F. Ruiz, F. Garcia, L. Calahorra, and C. Llorente, et al, "Business Process Modeling in Healthcare", *Studies in Health Technology and Informatics*, 2012, Vol. 179, pp. 75-87

[17] S. Hirano and S. Tsumoto, "Guide to Hepatitis Data for ECML/PKDD 2005 Discovery Challenge", retrieved on 05/02/2015 from http://lisp.vse.cz /challenge/index.html

[18] C.W. Gunther, H.M.W. Verbeek, "XES-Standard Definition. v. 2.0", 2014

[19] C.W. Gunther, "penXES Developers Guide. 1.0 RC5", 2009

[20] H.M.W. Verbeek, J.C.A.M. Buijs, B.F. van Dongen, and W.M.P van der Aaist, "XES, XESame and Prom 6", *Lecture Notes in Business Information Processing*, 2010, Vol. 72, pp. 60-75

[21] A.J.M.M Weijters, W.M.P. van der Aalst, A.K. Alves de Medeiros, "Process Mining with the Heuristics Miner-Algorithm", *Technology University Eindhoven Tech. Rep.*, Eindhoven 2006, WP 166, pp. 1-34

[22] V. Pieterse and P.E. Black, "Levenshtein distance.: Algorithms and Theory of Computation Handbook", retrieved on 05/03/2015 from http://www.nist.gov/dads/HTML/Levenshtein.html

[23] J.J.L Sander, D. Fahland, W.M.P. van der Aalst, "Process and Deviation Exploration with Inductive Visual Miner", BPM Demos 2014

[24] C.W. Gunther and W.M.P. Van der Aalst, Fuzzy Mining Adaptive Process Simplification Based on Multi-Perspective Metrics, *Lecture Notes in Computer Science*, 2007, Vol. 4714, 328-343

[25] W.M.P. van der Aalst, H. Reijers, and M. Song, "Discovering Social Networks from Event Logs", *Computer Supported Cooperative Work*, 2007, Vol. 14 No.6, pp. 549-593