A multi-step process mining approach based on the markov transition matrix

A. Hong Li¹, B. Kunman Li², and C. Lin Liu³, and D. Shaowen Yao⁴

¹School of Information Science and Engineering, Yunnan University, Kunming, Yunnan, China
²School of Computing and Engineering, University of Huddersfield, Queensgate, Huddersfield
³School of information, Yunnan Normal University, Kunming, Yunnan, China
⁴National Pilot School of software, Yunnan University, Kunming, Yunnan, China

Abstract - According to the situation that many workflow instances may deviate from the predefined model, this paper proposed a new process mining approach based on analyzing the workflow log to realize the workflow process reconstruction. First, build the markov transition matrix based on the workflow log, then design a multi-step process mining algorithm to mine the structural relationships between the activities, finally, this paper verifies the feasibility and applicability of the approach through a simulation example.

Keywords: process mining; markov transition matrix; process reconstruction

1 Introduction

With the development of information technology, more and more enterprises use workflow management systems (Workflow Management System, WFMS) to realize business process automated execution both inside and outside the enterprise. WFMS is driven by a predefined workflow model. Once the workflow model is been established, it usually won’t change in a very long period of time. However, due to the changing environment, the workflow instances which are the actual execution processes of the workflow often change. For example, during the process execution, companies will confer certain privileges to add executives or skip some of the nodes, even reset the given process transfer conditions, over time, the actual implementation of workflow instances will deviate from the predefined workflow model. If this change has not been detected in the long time, and the workflow instance is still driven by the old predefined model, then the efficiency of enterprises must have been affected. Based on this consideration, in order to improve the accuracy of workflow execution, modelers must be regularly informed of the actual implementation of the workflow, reconstruction or recommend better workflow model which reflects the actual needs of enterprises. In real life, each execution of the workflow instances is recorded in the WFMS log database in the form of workflow logs, which provide a solid data foundation for analyzing the changes in real execution workflow instance. In recent years, there exists a new process modeling idea - process mining, which extracts the structural description from the execution process collection [1]. Process mining is a new application of data mining, which is reproducing the real process of the business through analyzing workflow execution logs [2].

2 Process mining algorithms

The idea of process mining first appeared in the field of software engineering, was proposed by Joan-than E.Cook in 1995 [3]. In 1998, Agrawal first applied process mining technology to enterprise business process modeling [1]. Based on the mining process, workflow model reconstruction can be divided into single-step reconstruction and multi-step reconstruction [4]. Single-step reconstruction method is directly mined the workflow model based on the activities dependencies which explicit exists in the log. The single-step reconstruction algorithms include directed graph -based mining methods and WF-net based α algorithms proposed by Aalst. α algorithm is one of the classic algorithms of process mining, the mining process is simple, and the computation time is short, but the log noise handling capacity is insufficient, which is not suitable for mining complex structures workflow model such as non-free choice, loop, hidden activities and duplication activities [5]. A lot of research has been done on the improving of α algorithm, such as α* algorithm and β algorithm [6-7]. Multi-step reconstruction method adds the workflow log preprocessing or workflow model evaluation process, which makes the mining with higher accuracy, but the algorithm execution time is longer. Multi-step reconstruction methods include region-based mining method, clustering based mining method, genetic algorithm based mining method, frequency/dependency based mining method, multi-model mining and incremental mining methods [4, 8-15]. These algorithms all required of the integrity of the workflow model, which means that the mined workflow model needs to meet all the log, so the workflow model mined by these algorithms always with lower accuracy.

Process mining has attracted wide attention in academic domain, but there still lack of the algorithms with strong effectiveness for a wide range, good robustness and high efficiency. Some of the algorithms can only identify simple business process structure from the log due to the constraints of formal representation capacity, and some of the mining
algorithms have higher requirements of the log, which is difficult to handle the incomplete information or the noise. Consider of this situation, we designed a multi-step process mining method that first analyzes the workflow log transition probabilities between activities to build the Markov transition matrix, then mines the process logic relationship by defining a set of rules of logical relational mining, finally, we design the process mining algorithm to establish the actual structure relationship between the activities in order to reconstruct the workflow. At last, we use an elevator company’s standard contract process for example to verify the feasibility and applicability of the method.

3 Design of multi-step process mining method

In the enterprise business execution process, the workflow model can be seen as a stochastic process from one activity to another activity state which is a kind of Markov process or Markov chain. Because of the limited activity states, and the process activity to be performed in time “t+1” only related with the activity state in time “t”, so it is a finite-state Markov chain. This paper represents this process in \( \{x(t), t \in T \} \), \( x(t) \) is the process state in time “t”, \( p_{ij} \) is the one step transfer probability from \( x_i \) to \( x_j \), and \( p_0 \) has no relationship with the time “t”.

\[
p_{ij} = P[x(t+1) = x_j | x(t) = x_i] (x_i, x_j \in X)
\]

(1)

The matrix consists of all the transfer probability is called the transition probability matrix.

This paper puts forward a multi-step process mining algorithm based on the Markov transition matrix, and the algorithm process is shown in Fig.1.

![Fig.1. The main steps of the algorithm](image)

3.1 Preparation and preprocess of the log

Workflow log usually records the actual implementation process of the workflow model, and the log usually made up with the workflow instance name ‘Case_id’, activity name ‘Activity’, performer ‘Performer’ (can be specific people, or the application program), and execution time ‘Time’, etc. Among these, ‘Case_id’ used to identify execute times of one workflow, such as Case_1, Case_2, …: ‘Activity’ used to identify a specific activity of the workflow process, like x0, x1, …; ‘Performer’ and ‘Time’ are used to represent the specific actors and execution time of the activity.

Because of the input errors, there may be records missing, duplicate records and other reasons which may cause the noise or workflow logs incomplete. For the log with noise, it can do the noise filtering by setting the frequency threshold ‘0’. For incomplete logs, it can be filtered using the following two methods: a) list the sets of the end events of the log, if an instance’s end event does not belong to the set, then the instance logs are incomplete and should be removed; b) in the log, if a task is only has the start event without a corresponding end event, or only has the end event without a corresponding start event, then the instance is also incomplete and should be removed.

3.2 Construction of the markov matrix

The main steps are as follows:

Step 1: Compute the workflow instance numbers based on the log, which is means the executions times of workflow, make it ‘n’.

Step 2: Analyze the workflow instance, statistic all of the possible activity, i.e. \( X = \{x_0, x_1, \ldots, x_k\} \).

Step 3: Compute the transition possibility between the activity, make is \( p_{ij}= m_j/n \), \( m_j \) means the one step transition times from activity \( x_i \) to \( x_j \) in the n workflow instances.

Step 4: Establish the workflow transition matrix \( P \).

3.3 Definition of the logic relationship mining rules

Workflow process model usually consists of sequential structure, And-split/join, OR-split/join, loop structure (self-loops and multi-step cycle), which is composed of a variety of structural and process logic, includes sequence relationship, causal relationship, selection relationship, synchronous relationship and circular relationship, respectively use symbol ‘->’, ‘→’, ‘\&\&’, ‘|’, ‘\&\&’ to represent. This paper aims to derive the logical relationship in the workflow process through the analysis of the transition matrix \( P \). \( X_S \) represents the start of the process, \( X_E \) represents the end of the process, \( X \) represents the sets of process nodes, \( x_i \) represents the activity node in the process, \( p_{ij} \) represents the transition possibility between process nodes \( x_i \) and \( x_j \).

The logic relationship mining rules are as follows:

(a) Rule1: Identify the start of the process \( X_S \)
\[ \exists x_i \in X, \text{and} \forall x_i \in X, \text{if} p_{ij} = 0, \text{then} X_S = x_i \]

(b) Rule2: Identify the end of the process \( X_E \)
\[ \exists x_i \in X, \text{and} \forall x_j \in X, \text{if} p_{ij} = 0, \text{then} X_E = x_i \]

(c) Rule3: Mining rules of the sequence relationship
\[ \forall x_i, x_j \in X, \text{if} p_{ij} = 0, \text{then} x_i \rightarrow x_j \]

(d) Rule4: Mining rules of the causal relationship
\[ \forall x_i, x_j \in X, \text{if} p_{ij} = 1, \text{and} x_i \neq x_j \text{then} x_i \rightarrow x_j \]

(e) Rule5: Mining rules of the synchronous relationship
(And-Split, And-join), \( \forall x_i, x_{ij} \in X \).

Rule 5.1:
Rule 5.2: if \( p_{n1} = p_{n2} = \ldots = p_{nk} = \frac{1}{k} \neq 0 \), and \( x_i \neq x_{nk} \), then \( \{ x_{n1} / / x_{n2} / / \ldots / / x_{nk} \ ; \ x_i > x_{nk} \} \) is the And-Split activity node.

Rule 5.2: if \( p_{n1} = p_{n2} = \ldots = p_{nk} = \frac{1}{k} \neq 0 \), and \( x_i \neq x_{nk} \), then \( \{ x_{n1} / / x_{n2} / / \ldots / / x_{nk} \ ; \ x_i > x_{nk} \} \), \( x_i \) is the And-Join activity node.

(f) Rule 6: Mining rules of the selection relationship (OR-Split, OR-join)

\[ \exists x_i, x_{nk} \in X, p_{nk} > 0 \]

Then \( \{ x_{n1} # x_{n2} # \ldots # x_{nk}, x_i > x_{nk} \} \), \( x_i \) is the OR-Split activity node.

\[ \exists x_i, x_{nk} \in X, x_i \in S, \ and \ x_i \neq x_{nk} \]

\[ \{ x_{n1} # x_{n2} # \ldots # x_{nk} \ ; \ x_i > x_{nk} \} \], \( x_i \) is the OR-Join activity node.

(g) Rule 7: Mining rules of the self-circulation relationship

\[ \exists x_i \in X, x_{nk} \in X \setminus \{ x_i \} \]

if \( p_{il} = p \ (p \neq 1) \), then \{ \( \Diamond x_i; \ x_i > x_{nk} \) \}

(h) Rule 8: Mining rules of the multi-step cycle relationship

Rule 8.1: Mining the multi-step cycle activity node

\[ \forall x_i, x_{nk} \in X, x_i \neq x_{S}, x_i \neq X_E \]

if \( p_{ij} = 0, p_{ij} > 0 \) and \( \exists k (k \leq n - 1) \), then \( p_{ij}^{(k)} > 0 \)

Then \{ \( \{ x_{i} \ ; \ x_{i} > x_{n} \} \ ; \ x_{i} > x_{nk} \} \}

Refunction()

The main steps of function \( \text{Refunction}() \) are as follows:

Do \{
\[ \text{for} \ (\eta \leq n + 1, \ \eta = 0, \eta + +) \]
\[ \text{if} \ p_{i}^{(k-1)} > 0 \text{ and } p_{ij} > 0 \]
\[ \text{then} \{ X_{re} = X_{re} + \{ x_{\eta}; \ x_{\eta} > x_{ji} \} \}

k = k - 1; j = \eta
\}

While (\( k \geq 2 \)) the activities in \( X_{re} \) has cycle relationship, i.e. \( x_{i} \Diamond x_{i} \Diamond x_{n1} \Diamond \ldots \Diamond x_{nk} \)

Rule 8.2: Identify the start \( X_{re} \) and end \( X_{re} \) of the multi-step cycle

\[ \exists x_{nk} \in X_{re}, x_i \in X - X_{re} \text{ and } p_{nk} > 0 \]

Then \( \{ X_{re} = x_{nk}; x_i > X_{re} \} \)

\[ \exists x_{nk} \in X_{re}, x_i \in X - X_{re} \text{ and } p_{nk} > 0 \]

Then \( \{ X_{re} = x_{nk}; X_{re} > x_{i} \} \)

3.4 Design of the process mining algorithm

Based on the workflow process logical relationship mining rules Rule 1~Rule 8, we design the workflow mining algorithm Process \( (P, X) \). The algorithm can reconstruct the structural relationship between all the activities, and establish the ‘W’, ‘W_{and}’ and ‘W_{select}’ set.

Input: the Markov matric \( P \), the activities set \( X \), \( X = \{ x_0, x_1 \ldots x_n \} \)

Output: \( W \) is the set of the activities which have causal relationship, sequence relationship and cycle relationship;

\( W = \{(x_i, x_j) | x_i \rightarrow x_j \ or \ x_i > x_j \ or \ x_i \Diamond x_j \} \); \( W_{and} \) is the set of the activities which have synchronous relationship, \( W_{and} = \{(x_{n1}, x_{n2}, \ldots) | x_{n1} / / x_{n2} / / \ldots \} \); \( W_{select} \) is the set of the activities which have selection relationship, \( W_{select} = \{(x_{n1}, x_{n2}, \ldots) | x_{n1} # x_{n2} # \ldots \} \);

The main steps of the mining algorithm \( \text{Process} (P, X) \) are as follows:

**Initialization:**

\[ W = \emptyset, W_{and} = \emptyset, W_{select} = \emptyset \]

**Step 1:**

\[ \text{for} \ (i \leq n + 1, i = 0, i + +) \]

\[ \{
\[ \forall x_i \in I, \]
\[ \text{If Rule 1, then } X_S = x_i; \]
\[ \text{If Rule 2, then } X_E = x_i; \]
\[ \}

**Step 2:**

\[ \text{for} \ (i \leq n + 1, i = 0, i + +) \]

\[ \text{for} \ (j \leq n + 1, j = 0, j + +) \]

\[ \{
\[ \text{if Rule 3, then } W = W + \{(x_i, x_j) | x_i \rightarrow x_j \}; \]
\[ \text{if Rule 4, then } W = W + \{(x_i, x_j) | x_i \rightarrow x_j \}; \]
\[ \}

**Step 3:**

\[ \text{for} \ (i \leq n + 1, i = 0, i + +) \]

\[ \{
\[ \text{if } \exists x_{n1}, x_{n2} \ldots x_{nk} \in X, \ and \ Rule 5.1 \]
\[ \text{then } W_{and} = W_{and} + \{(x_{n1}, x_{n2}, \ldots x_{nk}) | x_{n1} / / x_{n2} / / \ldots x_{nk} \}; \]
\[ W = W + \{(x_i, x_{nk}) | x_i > x_{nk} \}; \]
\]
if \( \exists x_{n_1}, x_{n_2}, \ldots, x_{n_k} \in X \), and Rule 5.2
then \( W_{\text{and}} = W_{\text{and}} + \{(x_{n_1}, x_{n_2}, \ldots, x_{n_k})|x_{n_1} / \ldots / x_{n_k}\} \). \( W = W + \{(x_{n_k}, x_i)|x_{n_k} > x_i\}\);

if \( \exists x_{n_1}, x_{n_2}, \ldots, x_{n_k} \in X \), and Rule 6
then
\( W_{\text{select}} = W_{\text{select}} + \{(x_{n_1}, x_{n_2}, \ldots, x_{n_k})|x_{n_1} \neq x_{n_2} \ldots \neq x_{n_k}\} \);
\( W = W + \{(x_{n_k}, x_i)|x_{n_k} > x_i\}\);

if Rule 7
then \( W = W + \{(x_{n_1}, x_i)|x_{n_1} > x_i\} + \{(x_{n_1}, x_{n_k})|x_{n_1} > x_{n_k}\} \)

Step 4:
for \( (i \leq n + 1, i = 0, i + +) \)

\( t = t + 1; \)

\( W' = W' - \{(x_S, x_j)\}; I_j = I_j + \{x_j\} \)

\( \text{if } (x_S, x_j) \in W', \text{and } x_S \neq x_j \)
then \{direct \( x_S \) and \( x_j \) use double arrow,
there is self
- circulation relationship between \( x_S \) and \( x_j \} \)
\( t = t + 1; \)

\( W' = W' - \{(x_S, x_j)\}; I_j = I_j + \{x_j\} \)

\( \text{if } t = 1 \text{ then } X_S = x_j; \)
there is only one node \( x_j \) connect with \( X_S \)
\( \text{if } t = 1 \text{ then } \{ \text{Do } x_j = I_j; X_S = x_j \} \)
\( \text{DirectedNet} (X, W') \)
\( I_j = I_j - \{x_j\} \)
\( \text{While } I_j \neq \emptyset \}
\( \text{While } X_S = X_E \)

The process of the algorithm \( \text{DirectedNet} (X, W') \) is a cyclic process, which use self-circulation method to connect the active nodes in \( W' \).

The algorithm \( \text{DirectedNet} (X, W') \) first direct the start node \( X_S \) and the other nodes which has logical relationship with \( X_S \), and then based on the known logical relationship between activities, connect all the activity nodes use a single or double arrow until complete the whole direct network graph.

5 Simulation

To verify the feasibility of the method, this paper chooses a WFMS process (‘Pro’) log from one enterprise WFMS as the simulation example (see Table 1).

The log has 52 instances in total, Table 1 only lists the instance of Case 1– Case 4.

<table>
<thead>
<tr>
<th>Ca_id</th>
<th>Ac</th>
<th>Ca_id</th>
<th>Ac</th>
<th>Ca_id</th>
<th>Ac</th>
<th>Ca_id</th>
<th>Ac</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case_1</td>
<td>A</td>
<td>Case_2</td>
<td>D</td>
<td>Case_2</td>
<td>E</td>
<td>Case_2</td>
<td>J</td>
</tr>
<tr>
<td>Case_1</td>
<td>B</td>
<td>Case_3</td>
<td>D</td>
<td>Case_1</td>
<td>H</td>
<td>Case_4</td>
<td>E</td>
</tr>
<tr>
<td>Case_2</td>
<td>A</td>
<td>Case_1</td>
<td>F</td>
<td>Case_1</td>
<td>J</td>
<td>Case_3</td>
<td>G</td>
</tr>
<tr>
<td>Case_3</td>
<td>A</td>
<td>Case_1</td>
<td>F</td>
<td>Case_3</td>
<td>E</td>
<td>Case_4</td>
<td>F</td>
</tr>
<tr>
<td>Case_1</td>
<td>C</td>
<td>Case_4</td>
<td>B</td>
<td>Case_2</td>
<td>F</td>
<td>Case_3</td>
<td>I</td>
</tr>
<tr>
<td>Case_3</td>
<td>B</td>
<td>Case_5</td>
<td>A</td>
<td>Case_3</td>
<td>F</td>
<td>Case_4</td>
<td>G</td>
</tr>
<tr>
<td>Case_1</td>
<td>D</td>
<td>Case_2</td>
<td>C</td>
<td>Case_4</td>
<td>D</td>
<td>Case_4</td>
<td>H</td>
</tr>
<tr>
<td>Case_2</td>
<td>B</td>
<td>Case_3</td>
<td>C</td>
<td>Case_2</td>
<td>G</td>
<td>Case_3</td>
<td>J</td>
</tr>
<tr>
<td>Case_4</td>
<td>A</td>
<td>Case_4</td>
<td>C</td>
<td>Case_2</td>
<td>I</td>
<td>Case_4</td>
<td>J</td>
</tr>
<tr>
<td>Case_1</td>
<td>E</td>
<td>Case_1</td>
<td>G</td>
<td>Case_3</td>
<td>F</td>
<td>Case_3</td>
<td>J</td>
</tr>
</tbody>
</table>

The steps of the mining process are as follows:

1) Build the Markov transition matrix of ‘Pro’
a) the process of ‘Pro’ has 52 instances in all, i.e. n=52;
b) the process of ‘Pro’ has 10 activities in total (from A to J), X = {A, B, ... J}
c) compute the one-step transition possibility between the activities, 
\[ p_{ij} = \frac{m_{ij}}{n} \text{, e.g. } p_{AB} = \frac{m_{AB}}{52} = \frac{52}{52} = 1; \]
d) Establish the Markov transition matrix P
\[
P = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\]
e) According to the relationship sets (W, W_{and} and W_{select}) and the activity set X, the directed network graph model of ‘Pro’ is shown in Fig.2.

![Diagram](https://example.com/diagram.png)

Fig.2. The direct network graph model of ‘Pro’

### 6 Conclusion

Since the changing environment internal and external of the enterprise, the workflow instances may deviate from the predefined models, so modelers need to reconstruct the model according to the actual situation. To overcome the limitation of the existing process mining algorithms, this article designed a multi-step process mining method based on Markov transition matrix, by analyzing the workflow log, automatically deduced the actual structure of the relationship between activities, thus implement the workflow reconstruction. Finally, a practical simulation case study shows that the method is feasible and applicable. Follow-up studies will focus on how to automatically derive sub-processes, as well as how to automatically mining rename tasks in order to improve and enhance the mining ability of the method.

### 7 References


