Dealing With Concept Drifts in Process Mining

**ABSTRACT**

Although most business processes change over time, contemporary process mining techniques tend to analyze these processes as if they are in steady-state. Processes may change suddenly or gradually. The drift may be periodic (e.g. due to seasonal influences) or one-of-a- kind (e.g., the effects of new legislation). For process management it is

crucial to discover and understand such concept drifts in processes.

**EXISTING SYSTEM:**

The process is stable and enough example traces have been recorded in the event log, it

is possible to discover a high quality process model that can be used for performance analysis, compliance checking, and prediction. Unfortunately, most processes are not in steady-state. In today's dynamic marketplace, it is increasingly necessary for enterprises to streamline their processes so as to reduce costs and to improve performance.

**PROPOSED SYSTEM:**

The proposed four features characterizing the control flow dependencies between activities. These features are shown to be effective in detecting process changes. An event log can be transformed into a data set D, which can be considered as a time series by these features. Change detection is done by considering a series of successive populations1 of feature values and investigating if there is a significant difference between two successive populations. The premise is that differences are expected to be perceived at change points provided appropriate characteristics of the change are captured as features.

**MODULE DESCRIPTION:**

# **Number of Modules**

After careful analysis the system has been identified to have the following modules:

1. **Change Point Detection Module.**
2. **Change Localization and Characterization Module.**
3. **Change Process Discovery Module.**

**1. Change Point Detection Module:**

The first and most fundamental problem is to detect concept drift in processes, i.e., to detect that a process change has taken place. If so, the next step is to identify the time periods at which changes have taken place. For example, by analyzing an event log from an organization (deploying seasonal processes), one should be able to detect that process

changes happen and that the changes happen at the onset of a season.

**2.Change Localization And Characterization Module:**

Once a point of change has been identified, the next step is to characterize the nature of change, and identify the region(s) of change (localization) in a process. Uncovering the nature of change is a challenging problem that involves both the identification of change perspective (e.g., controflow, data, resource, sudden, gradual, etc.) and the identification of the exact change itself. For example, in the example of a seasonal process, the change could be that more resources are deployed or that special offers are provided during holiday seasons.

**3. Change Process Discovery Module:**

Having identified, localized, and characterized the changes, it is necessary to put all of these in perspective. There is a need for techniques/tools that exploit and relate these discoveries. Unraveling the evolution of a process should result in the discovery of the change process describing the second order dynamics. For example, in the example of a seasonal process, one could identify that the process recurs every season. Also, one can show an animation on how the process evolved over a period of time with annotations showing several perspectives such as the performance metrics (service levels, throughput time, etc.) of a process at different instances of time.

**SOFTWARE REQUIREMENTS**:

Operating System : Windows

Technology : Java and J2EE

Web Technologies : Html, JavaScript, CSS

IDE : My Eclipse

Web Server : Tomcat

Tool kit : Android Phone

Database : My SQL

Java Version : J2SDK1.5

**HARDWARE REQUIREMENTS**:

Hardware : Pentium

Speed : 1.1 GHz

RAM : 1GB

Hard Disk : 20 GB

Floppy Drive : 1.44 MB

Key Board : Standard Windows Keyboard

Mouse : Two or Three Button Mouse

Monitor : SVGA