STAGE M2RI (2007-2008)

Concept change detection and model adaptation for chronicle recognition based online diagnosis

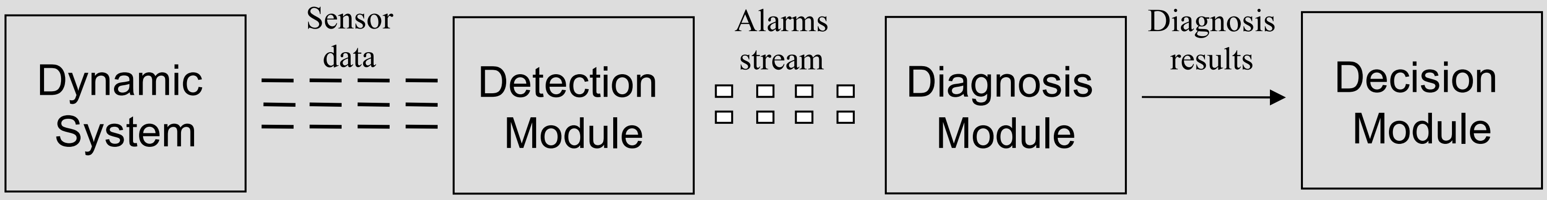
Presenter : Qasim Malik

Supervisors : René Quiniou, DREAM
Thomas Guyet, DREAM

UNIVERSITÉ DE RENNES 1

Online diagnosis

- Online diagnosis consists in identifying the components failure within a dynamic system in real time
- Requires a monitoring system to identify abnormal situations as quickly as possible
- Typical components of a monitoring system
  - Detection Module
  - Diagnosis Module (model-based)
  - Decision Module



•Telecommunication system

•Patient in intensive care unit

Alarms generation

Identification of abnormal situations

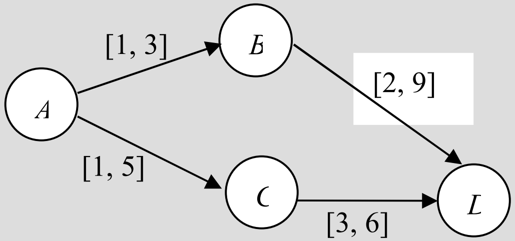
Propose actions

Diagnosis problem with concept change

- Concept change refers to more or less radical changes in the target concept induced by changes in the hidden context not known explicitly
- Concept change occurs because
  - Observed system is dynamic
  - Data is of streaming nature
- Consequences
  - Unable to recognize all the occurrences of failures (current chronicles should be updated)
  - Reduce the effectiveness of diagnosis task
- Solution
  - Detect the concept change
  - Overcome it
    - Build a new model
    - Perform adaptations to the current model

Online diagnosis using chronicle recognition

- A model is represented by a set of chronicles: each failure is associated to one or several descriptions of abnormal situations (event sequences)
- A chronicle consists of events that are interlinked by temporal constraints
  - Events corresponds to the alarms issued by detection module
  - Represents a possible abnormal situation that can occur in the system



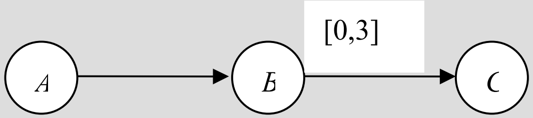
- The diagnostic task is to recognize the chronicles online
- Achieved by
  - Chronicle recognition system (CRS)
    - We have used the CRS software provided by France Telecom

States of a chronicle

- A chronicle model contains non-instantiated temporal variables
- A chronicle instance is a copy of a chronicle containing uninstantiated temporal variables
- A partially recognized chronicle is a chronicle instance where one or several temporal variables have been instantiated with the timestamp of the corresponding events that satisfies the related temporal constraints
- A recognized chronicle is a chronicle instance where all the temporal variables have been instantiated and all the temporal constraints are satisfied
- A chronicle instance is discarded when some of its awaited events could not arrive in time i.e. given the current time, the occurrence time of some event, according to the temporal constraints between this event and previously occurred events, is passed

Chronicle recognition

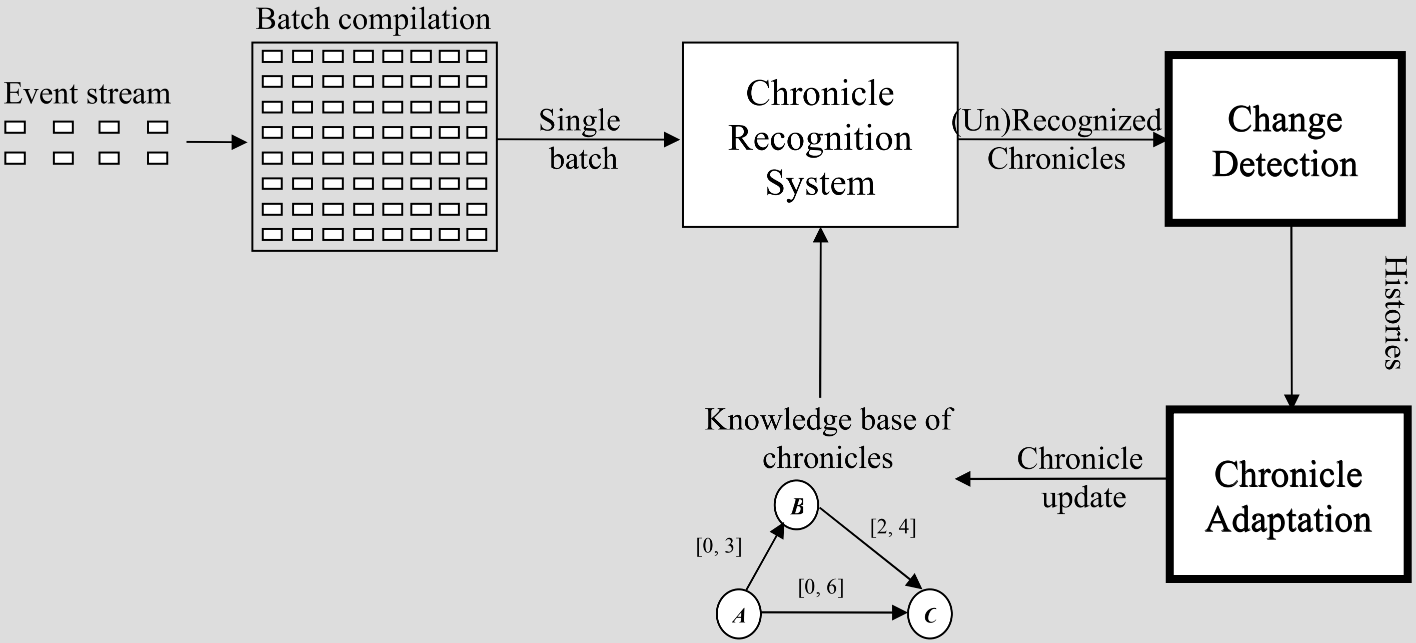
- Chronicle :



- Event stream : (A,0), (B3), (B,8), (C,11), ...

T	***Evt***	***Chronicle instances***	***Action***
0	**A**	**C1: {(A,0)}**	**Creation of instance C1**
3	**B**	**C1: {(A,0)}, C2: {(A,0), (B,3)}**	**Creation of instance C2**
6		**C1: {(A,0)}**	**Deletion of C2**
8	**B**	**C1: {(A,0)}, C3: {(A,0), (B,8)}**	**Creation of C3**
11	**C**	**C1: {(A,0)}, C3: {(A,0), (B,8)},**	**Creation of C4,**
		C4: {(A,0), (B,8), (C,11)}	**Recognition of C4**
12		**C1: {(A,0)}**	**Deletion of C3**
...			

Overall Process: data stream context

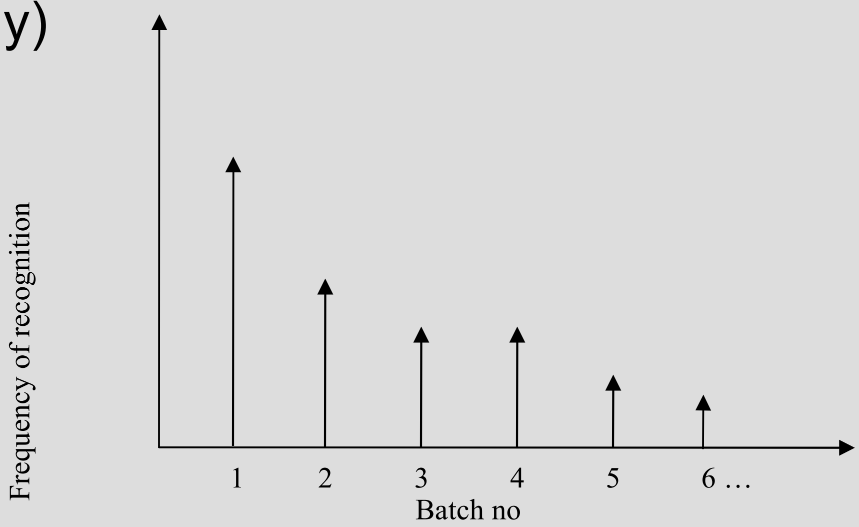


Concept change detection

- What is concept change?
  - When a situation occurs where there is a need for changing the definition of at least one chronicle
- Methods proposed for concept change detection
  - Frequency of chronicle recognition
    - Percentage of completion
    - Killings at each temporal constraint
  - Chronicle vs. sub-chronicle recognitions

Frequency of chronicle recognition

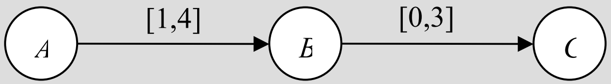
- Low frequency of recognition may be due to change (not necessarily)



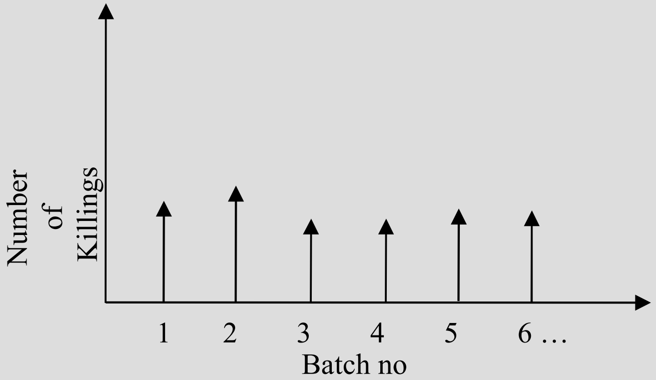
- Two possibilities for low frequency of chronicle recognition
  - Either a change has occurred
  - The failure associated with some chronicle is not occurring
- To separate these two, we introduce two measures
  - Percentage of completion
    - Offered a lot of shortcomings
  - Killings at each temporal constraint

Killings at each temporal constraint

- Find the number of killings of chronicle instances at each temporal constraint
- Notion of killings
  - Time stamped events
    - *(A, 0), (B, 3), (C, 11)*



- Need to maintain batch wise profile of killings for each temporal constraint

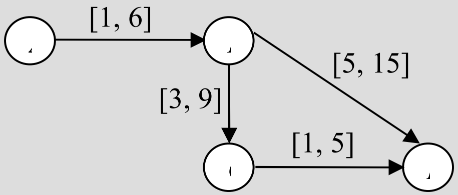


- **Low** frequency of recognition of chronicle and significant **increase** in number of killings at some temporal constraint

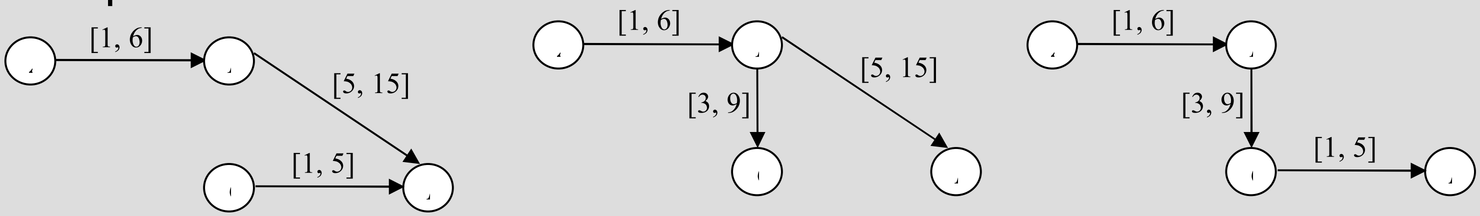
  ⟹ Change has occurred in the chronicle
- No functionality found in CRS that can find the temporal constraint responsible for killing the largest partial instance

Chronicle vs. sub-chronicle recognitions

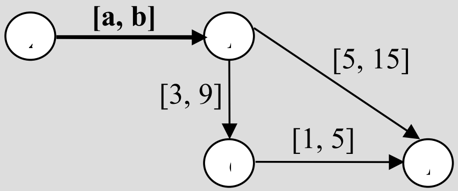
- Divide a chronicle into sub-chronicles by removing the temporal constraints one by one
- Chronicle



- Sub-chronicles
  - Two possible cases

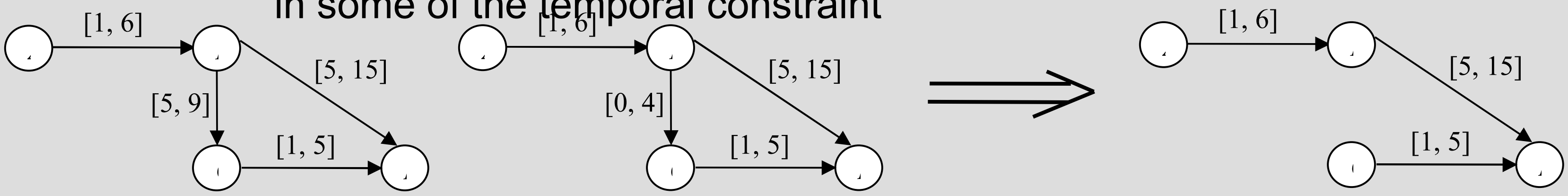


Need to have some idea about the extent of change



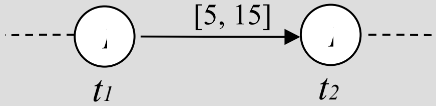
Chronicle vs. sub-chronicle recognitions

- Trigger for change detection
  - Whenever a sub-chronicle is recognized
  - Works because
    - Mutual exclusiveness is ensured between chronicle and its sub-chronicle recognitions
    - Probability of occurrence of sub-chronicle is very low in the absence of change
- Limitations
  - Change can occur in only one of the temporal constraint
  - The chronicles must not have significant similarity with each other
    - Significant similarity means similar organization of events with difference in some of the temporal constraint

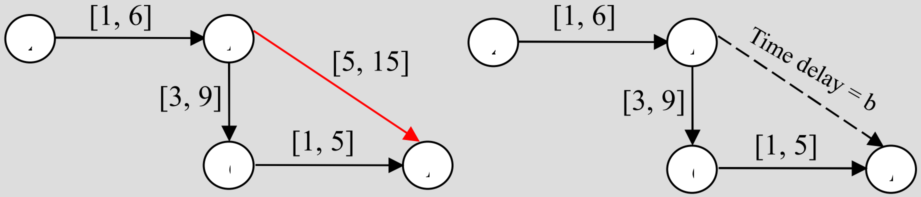


Model adaptation

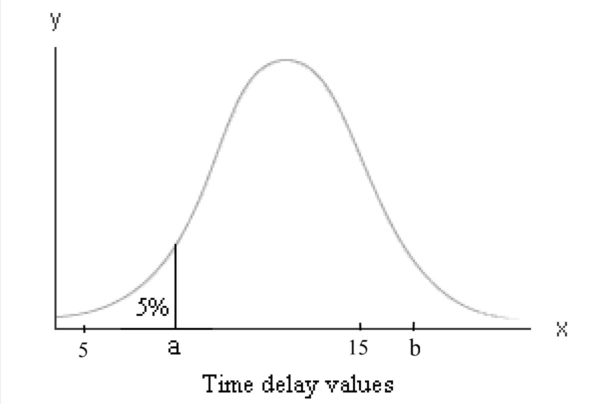
- Need to maintain some kind of history to overcome the detected change
- Interested in storing the history of ‘time delay’ values for each temporal constraint



  - $(t_2 - t_1)$ is the time delay value
- For performing adaptation
  - Find the time delay value in the recognized sub-chronicle to find one limit

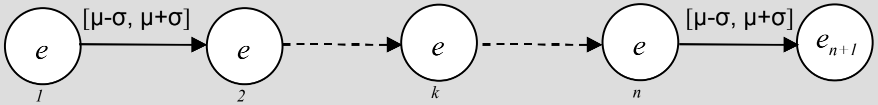


  - Fit the gaussian model on the recent time delay value data to find other limit



Implementation

- Chronicle recognition performed using CRS software
  - Requires two files as input
    - Chronicle models
      - Generated sequential chronicles using defined events



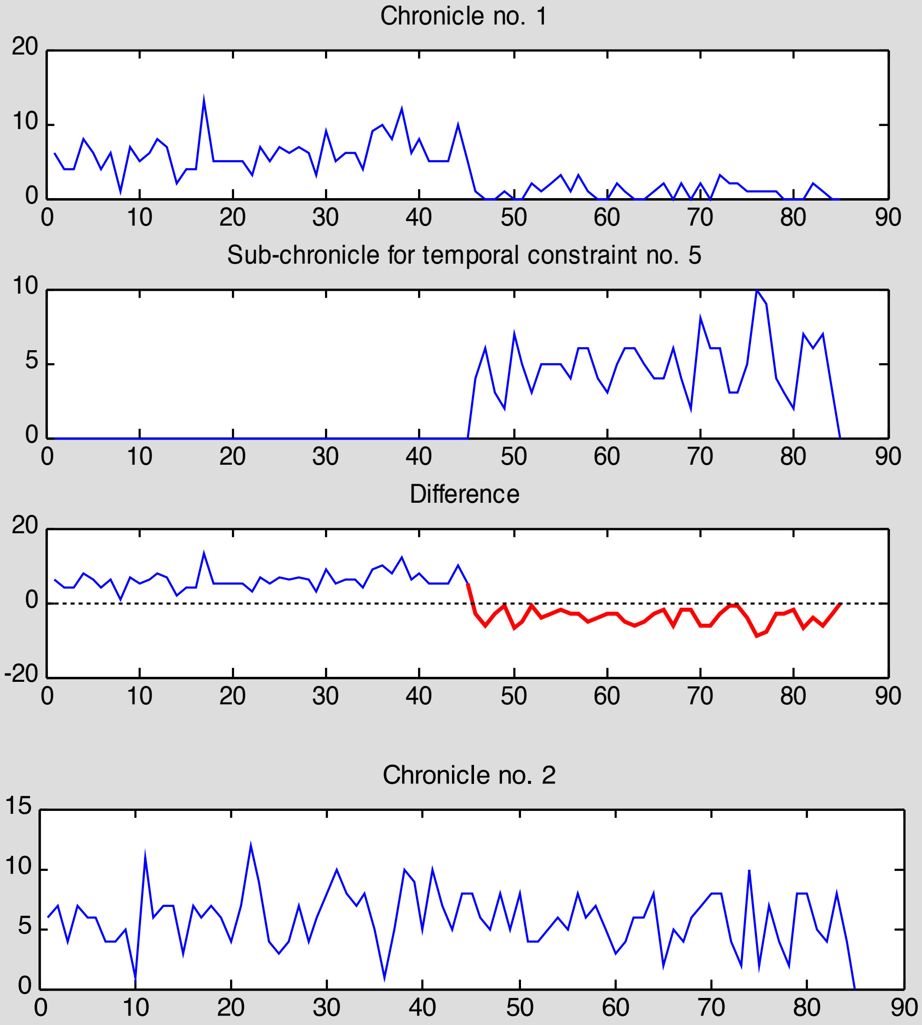
      - Specified mean, μ, and variation, σ, for each temporal constraint
    - Events stream data
      - Randomly inserted time stamped events between chronicles
      - Randomly specified a batch number for few randomly chosen chronicles from where the change will begin to occur
      - Change was introduced by shifting the temporal interval
- Change detection method: chronicle vs. sub-chronicle recognition
  - Generated sub-chronicles and wrote them in chronicle models file along with original chronicles

Experiment

- □ Number of distinct chronicles = 12
- □ Number of distinct events = 14
- □ Number of events in each chronicle ranges from 6-11
- □ Number of events in the file = 127500
- □ Number of chronicles occurrences that event file will contain = 4819
- □ Batch size = 1500
- □ Number of batches = 127500 / 1500 = 85

Chronicle no.	Change to be introduced?	Temporal constraint no.	Batch no.
1	Yes	5	46
2	No	-	-
3	Yes	7	39
4	Yes	1	47
5	No	-	-
6	No	-	-
7	No	-	-
8	Yes	3	39
9	Yes	6	39
10	Yes	4	35
11	Yes	2	55
12	Yes	2	56

Results

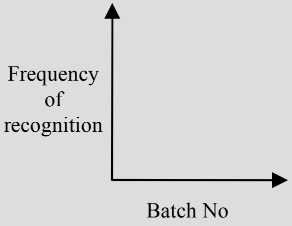


Chronicle no. 1

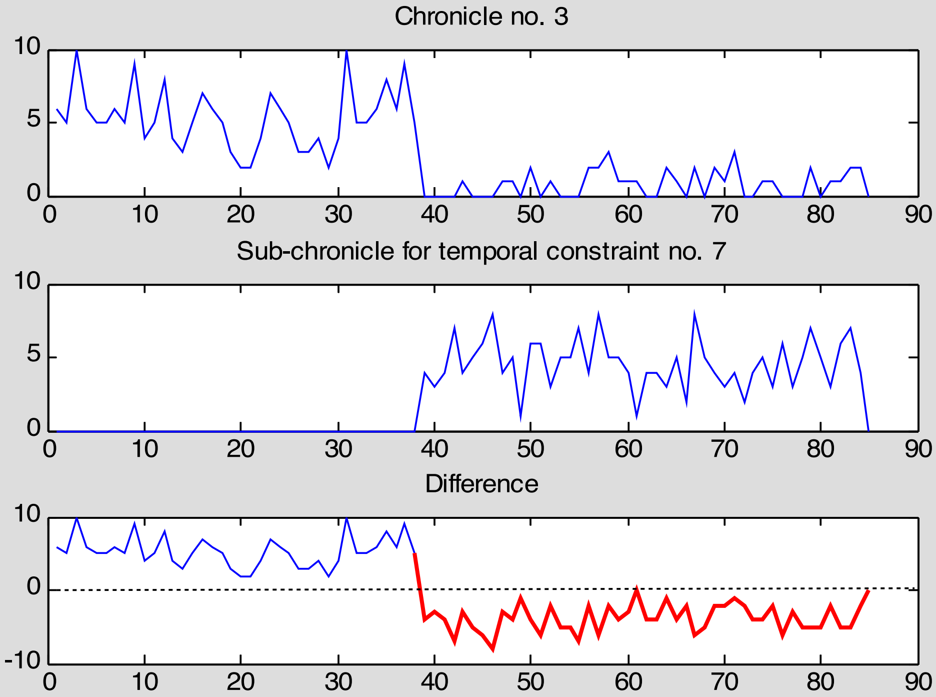
Sub-chronicle for temporal constraint no. 5

Difference

Chronicle no. 2

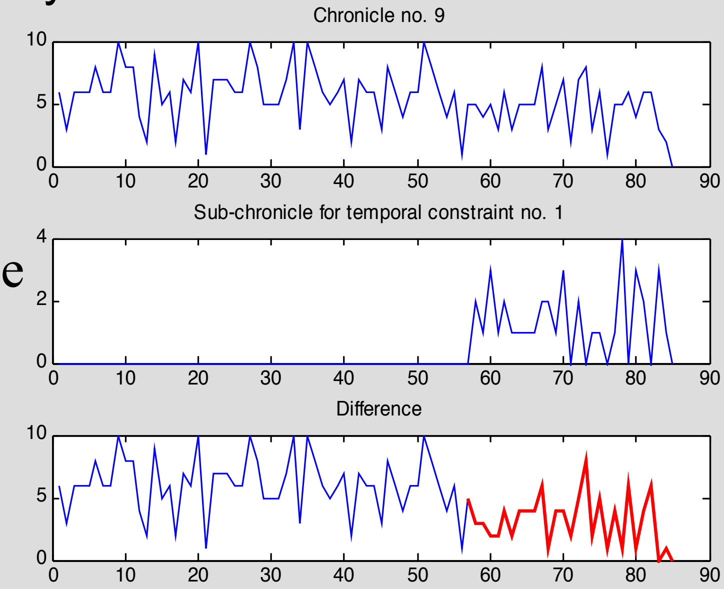


Results


Chronicle no. 3
10
5
0
0
10
20
30
40
50
60
70
80
90
Sub-chronicle for temporal constraint no. 7
10
5
0
0
10
20
30
40
50
60
70
80
90
Difference
10
0
-10
0
10
20
30
40
50
60
70
80
90

Discussion

- Two important assumptions were made
  - Idea about the extent of change
    - Case of disconnected sub-chronicles
  - Chronicles must not have significant similarities with each other
    - Criteria for change detection may not work
- Another criteria for change detection
  - When difference between chronicle and sub-chronicle recognitions falls below the zero line
  - Works only in case of significant change



Conclusion

- □ Introduced the notion of concept change
- □ Proposed few methods for concept change detection and model adaptation
- □ Presence of real life data set would have helped to discover different real life problems
- □ Need to work further to make the detection technique more generalized
- □ Chronicles similarity may be used in recognition process to make the detection method more efficient

Queries!!

???