

CS 6604 Assignment #4

Assigned: Nov 30, 2007

Date Due: Dec 10, 2007, 11:59:59pm, by email to the instructor

Read the paper [Laxman, Sastry, Unnikrishnan, “Discovering Frequent Episodes and Learning Hidden Markov Models: A Formal Connection”, IEEE TKDE, Nov 2005] with a view toward using the algorithm presented there as a ‘building block’ to mine more complex patterns (as given below) from a given event stream. See if the algorithm as described here already can mine the given pattern and, if not, how you will adapt/modify the algorithm from this paper. You are also welcome to experiment with the TDMiner software available at <http://neural-code.cs.vt.edu> which implements this algorithm. Specifically TDMiner already has facilities for mining the first three types of patterns below.

1. (5 points) Frequently occurring total orders (i.e., ‘something followed by something followed by something ..., in that order’)
2. (5 points) Frequently occurring anti-chains.
3. (10 points) Frequently occurring total orders with specified durations between the given events.
4. (20 points) Frequently occurring hammock posets.
5. (30 points) Frequently occurring leveled posets.
6. (30 points) Frequently occurring series-parallel posets.

The notions of hammock and leveled posets are defined in many places - for instance, see: <http://www.cs.vt.edu/~naren/papers/poset.pdf>. The definition of series-parallel posets can be obtained, for instance, from the paper [Mannilla, Meek, “Global Partial Orders from Sequential Data”, KDD 2000]. For each class of pattern, identify what tweaks you will need to make the basic algorithm of Laxman, Sastry, and Unnikrishnan.

Turn in a typed (not handwritten) paper copy giving answers to the questions, plots, including a brief description of how you solved each question. Write enough to convince us that you completed the assignment independently.