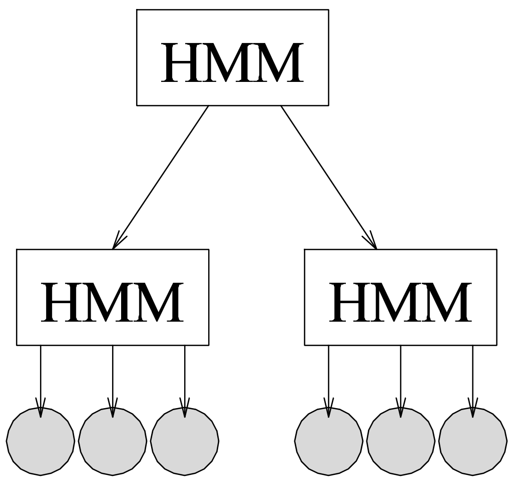
Anomaly Detection using Hierarchical Hidden Markov Models

Suratna Budalakoti
University of California Santa Cruz

Motivation



- Humans behavior often forms a hierarchy, where a number of smaller subtasks are performed with the aim of achieving a larger task.
- The Hierarchical Hidden Markov Model is a good model for this kind of behavior.

What is a Hierarchical Hidden Markov Model?

- The Hierarchical HMM is an extension of the HMM that is designed to model domains with hierarchical structure[MP2001].

- Special case of stochastic context free grammars (bounded depth of recursion).

- Common Application Areas: Natural language, speech, visual action recognition.

Anomaly Detection

- The term most commonly refers to anomaly detection with relation to computer systems.
  - Build a profile of normal usage (learning).
  - Deviations from the normal are flagged (inference).

- Common approaches:
  - Nearest Neighbor based approaches.
  - Hidden Hierarchical Models (HMMs).

Learning in HHMMs

- Data is Dense
  - Events are not are or hidden in a lot of noise.
    - e.g.: Modeling a soccer match.
  - Hill-climbing approach to learn the structure.
- Data is Sparse
  - Events are few and far-between.
  - Need a mining step to discover the events.

Learning Step for anomaly detection

- Bottom-up Approach:
  1. Find frequently occurring sequences in the data using a frequent sequence mining algorithm.
  2. Cluster these sequences into groups of similar sequences. Train an HMM over each group.
  3. Replace all members of the same group with an abstract symbol, to create a new set of sequences.
  4. Go to Step 1.

Frequent Sequence Mining

- The *Apriori* algorithm
  - Start with candidate frequent sequences of length 2.
  - Generate candidate sequences of length k from frequence sequences of length k-1.
- Example:
  - IsFreq(ABCD) & IsFreq(BCDE)

    => Test for ABCDE

Clustering Sequences into Groups (using edit distance)

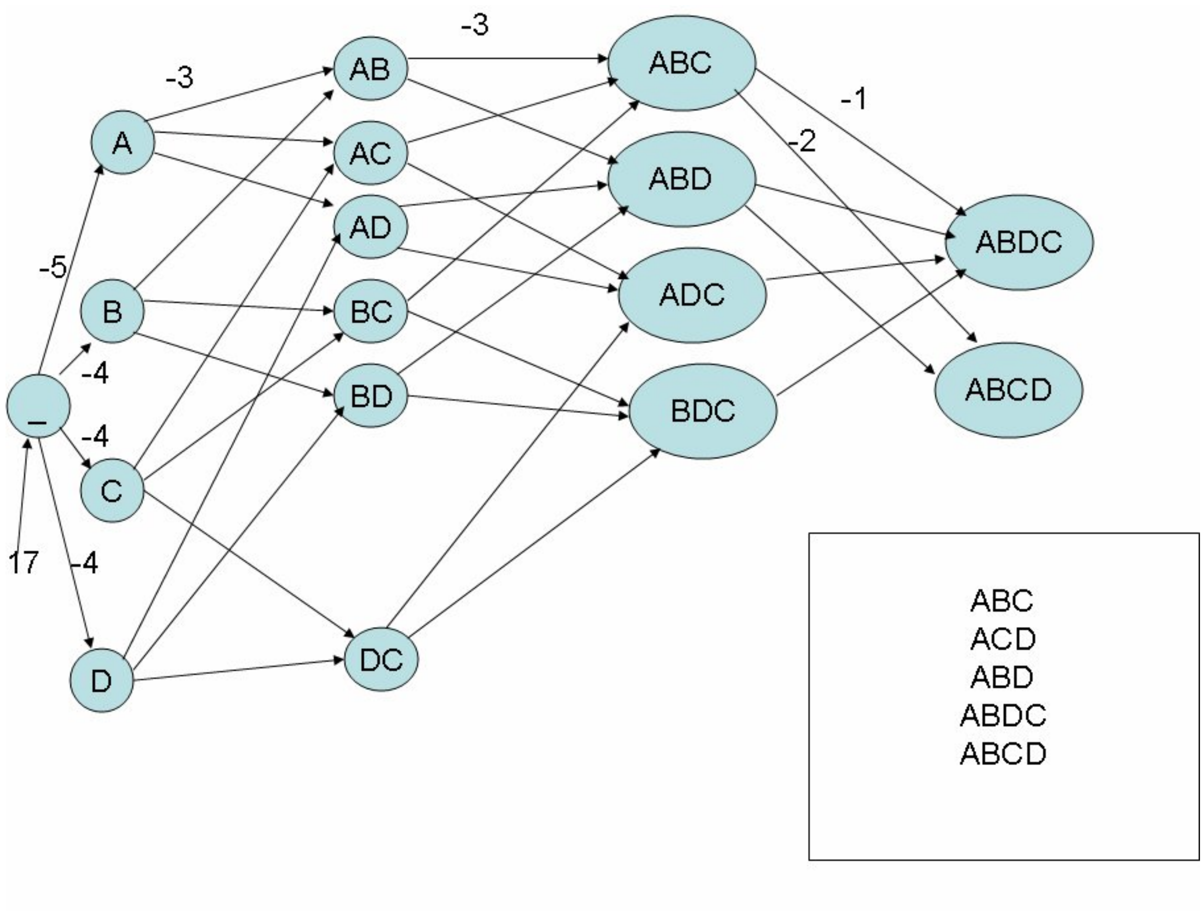
- Clustering sequences is always problematic.
  - Metric space where edit distance is embedded is not known.
  - Existing methods do not scale well.

- Possible approaches
  - Hierarchical clustering
  - The CLARA clustering algorithm
    - Sampling based approach.
    - Does not scale well as the number of clusters required increases.

Outline of clustering algorithm

- Select k arbitrary sequences as cluster centroids.
- Assign each sequence to the centroid it is closest to, using edit distance as a similarity measure.
- Calculate the consensus/centroid sequence for each cluster.
- Assign each sequence to the centroid it is closest to.
- If one or more sequences change clusters, go to step 3.
- Stop.

Constructing the Consensus Sequence



Inference in Hierarchical Hidden Markov Models

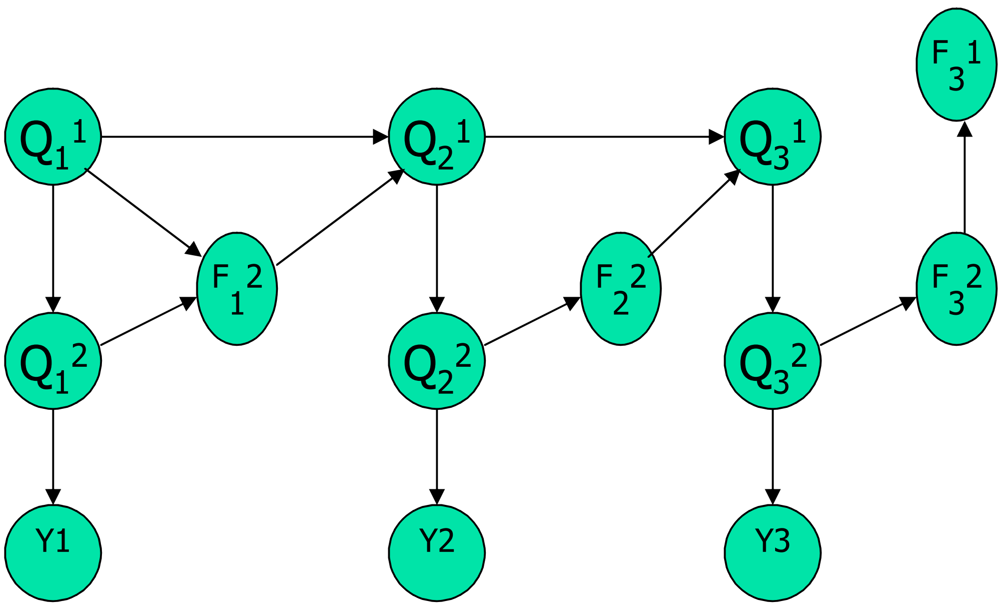
- Murphy and Paskin recently showed how to convert an HHMM into a Dynamic Bayesian Network (DBN).
- DBN: A DBN is a Bayesian Network built over a dynamic system.

DBNs (continued)

- Defined by a pair (B_1, B_{tran}).
  - B_1: Defines prior Z_1.
  - B_{tran}: Two-slice temporal Bayesian Network which defines

$$P(Z_t \mid Z_{t-1}) = \prod_{i=1}^{N} P(Z_t^i \mid Pa(Z_t^i))$$

An HHMM unrolled into a DBN (for three stages)



Inference

- Inference can now be performed as in a standard Bayesian Network.
  - Form a clique tree for the network.
  - Apply a forward-backward message passing algorithm.

Data Description

- Data collected from Unix logs of users at Purdue University.
  - 10 users (User 0 to User 9).
  - Approximately 2000 sequences per user.
  - Average length of sequences is around 80 commands.

Testing

- Both a Hidden Markov Model and a Hierarchical Hidden Markov Model was trained on the data, for User 0 and User 1.
- Aim was to compare accuracy for masquerade detection.
- A likelihood threshold was arbitrarily chosen for classification in each case.

Results

	User 0	User 1
User 0	66.2%	34.6%
User 1	31.4%	73.6%

	User 0	User 1
User 0	71.2%	36.4%
User 1	33.2%	74.85

Thank you