A Probabilistic Approach to Fast Pattern Matching in Time Series Databases

Eamonn Keogh and Padhraic Smyth*
Department of Information and Computer Science
University of California, Irvine
CA 92697-3425
{eamonn,smyth}@ics.uci.edu

Abstract

The problem of efficiently and accurately locating patterns of interest in massive time series data sets is an important and non-trivial problem in a wide variety of applications, including diagnosis and monitoring of complex systems, biomedical data analysis, and exploratory data analysis in scientific and business time series. In this paper a probabilistic approach is taken to this problem. Using piecewise linear segmentations as the underlying representation, local features (such as peaks, troughs, and plateaus) are defined using a prior distribution on expected deformations from a basic template. Global shape information is represented using another prior on the relative locations of the individual features. An appropriately defined probabilistic model integrates the local and global information and directly leads to an overall distance measure between sequence patterns based on prior knowledge. A search algorithm using this distance measure is shown to efficiently and accurately find matches for a variety of patterns on a number of data sets, including engineering sensor data from space Shuttle mission archives. The proposed approach provides a natural framework to support user-customizable "query by content" on time series data, taking prior domain information into account in a principled manner.

Introduction and Motivation

Massive time series data sets are commonplace in a variety of online monitoring applications in medicine, engineering, finance, and so forth. As an example, consider mission operations for NASA's Space Shuttle. Approximately 20,000 sensors are telemetered once per second to Mission Control at Johnson Space Center, Houston. Entire multi-day missions are archived at this 1 Hz rate for each of the 20,000 sensors. From a mission operations viewpoint, only a tiny fraction of the data can be viewed in real-time, and the archives are too vast to ever investigate. Yet, the data are potentially very valuable for supporting diagnosis, anomaly detection, and prediction. This is a familiar problem in archived time series storage: high-dimensional data sets at very high resolution make manual exploration virtually impossible.

In this paper we address the general problem of matching a sequential pattern (called a query Q) to a time series database (called the reference sequence R). We will assume that Q and R are each real-valued univariate sequences. Generalizations to multivariate and categorical valued sequences are of significant practical interest but will not be discussed here. To keep the discussion and notation simple we will assume that the sequence data are uniformly sampled (i.e., uniformly spaced in time). The generalization to the non-uniformly sampled case is straightforward and will not be discussed. The problem is to find the k closest matches in R to the query Q. Most solutions to this problem rely on three specific components: (1) a representation technique which abstracts the notion of shape in some sense, (2) a distance measure for pairs of sequence segments, and (3) an efficient search mechanism for matching queries to reference sequences. The contribution of this paper is primarily in components (1) and (2). A piecewise linear representation scheme is proposed and combined with a generative probabilistic model on expected pattern deformations, leading to a natural distance metric incorporating relevant prior knowledge about the problem.

Related Work

There are a large number of different techniques for efficient subsequence matching. The work of Faloutsos, Ranganathan, and Manolopolous (1994) is fairly typical. Sequences are decomposed into windows, features are extracted from each window (locally estimated spectral coefficients in this case), and efficient matching is then performed using an R^*-tree structure in feature space. Agrawal et al. (1995) proposed an alternative approach which can handle amplitude scaling, offset translation, and "don't care" regions in the data, where distance is determined from the envelopes of the original sequences. Berndt and Clifford (1994) use dynamic time-warping approach to allow for "elasticity" in the temporal axis when matching a query Q

*Also with the Jet Propulsion Laboratory 525-3660, California Institute of Technology, Pasadena, CA 91109.

to a reference sequence R. Another popular approach is to abstract the notion of shape. Relational trees can be used to capture the hierarchy of peaks (or valleys) in a sequence and tree matching algorithms can then be used to compare two time series (Shaw and DeFigueiredo, 1990; Wang, et al., 1994).

A limitation of these approaches in general is that they do not provide a coherent language for expressing prior knowledge, handling uncertainty in the matching process, or integrating shape cues at both the local and global level. In this paper we investigate a probabilistic approach which offers a theoretically sound formalism for

- Integration of local and global shape information,
- Graceful handling of noise and uncertainty, and
- Incorporation of prior knowledge in an intuitive manner.

The probabilistic approach to template matching is relatively well-developed in the computer vision literature. The method described in this paper is similar in spirit to the recent work of Burl and Perona (1996).

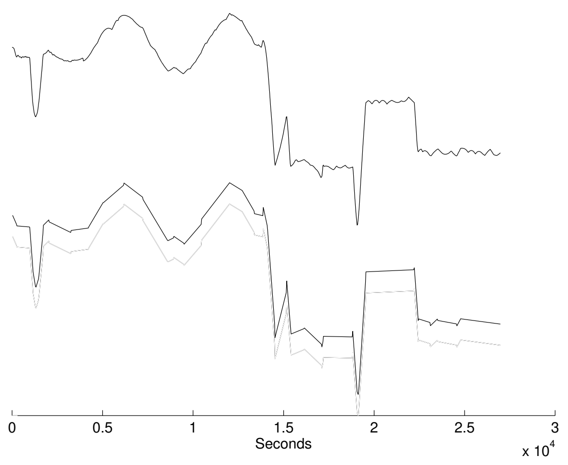

Figure 1: Automated segmentation of an inertial navigation sensor from Space Shuttle mission STS-57. (a) original data, the first 7.5 hours of the mission, originally 27,000 data points, (b) the segmented version of this sequence, $K = 43$ segments chosen by the multi-scale merging algorithm described in the text.

A Segmented Piecewise Linear Representation

There are numerous techniques for representing sequence data. The representation critically impacts the sensitivity of the distance measure to various distortions and also can substantially determine the efficiency of the matching process. Thus, one seeks robust representations which are computationally efficient to work with. Spectral representations are well-suited to sequences which are locally stationary in time, e.g., the direct use of Fourier coefficients (as in Faloutsos et al. (1995)) or parametric spectral models (e.g., Smyth, 1994). However, many sequences, in particular those containing transient behavior, are quite non-stationary and may possess very weak spectral signatures even locally. Furthermore, from a knowledge discovery viewpoint, the spectral methods are somewhat indirect. We are interested here in pursuing a representational language which can directly capture the notion of sequence shapes and which is intuitive as a language for human interaction.

There is considerable psychological evidence (going back to Attneave's famous cat diagram, 1954) that the human visual system segments smooth curves into piecewise straight lines. Piecewise linear segmentations provide both an intuitive and practical method for representing curves in a simple parametric form (generalizations to low-order polynomial and spline representations are straightforward). There are a large number of different algorithms for segmenting a curve into the K "best" piecewise linear segments (e.g., Pavlidis (1974)). We use a computationally efficient and flexible approach based on "bottom-up" merging of local segments into a hierarchical multi-scale segmentation, where at each step the two local segments are merged which lead to the least increase in squared error. Automated approaches to finding the best number of segments K can be based on statistical arguments (penalized likelihood for example or Minimum Description Length as in Pednault (1991) for this problem). We found that for piecewise linear segmentations, simple heuristic techniques for finding good values of K worked quite well, based on halting the bottom-up merging process when the change in approximation error in going from K to $K-1$ increased substantially. Figure 1 shows the segmentation of an inertial navigation sensor from the first 8 hours of a Space Shuttle mission by this method. For practical applications it may be desirable to have K chosen directly by the user to reflect a particular resolution at which matching is to be performed.

Probabilistic Similarity Measures

Defining "similarity" metrics is a long-standing problem in pattern recognition and a large number of distance measures based on shape similarity have been proposed in the literature. Typically these similarity measures are designed to have certain desirable properties such as invariance to translation or scaling.

Here we propose a probabilistic distance model based on the notion of an ideal prototype template which can then be "deformed" according to a prior probability distribution to generate the observed data. The model consists of *local* features which are then composed into a *global* shape sequence. The local features are allowed some degree of deformation and the global shape sequence has a degree of elasticity allowing stretching in time and amplitude of the signal. The degree of deformation and elasticity are governed by prior probability distributions.

Specifically, let Q be a query sequence consisting of k local features, i.e., $Q = \{q_1, \ldots, q_k\}$. For example, q_1 and q_3 could be peaks and q_2 could be a plateau. Let l_i, $1 \leq i \leq k-1$, be the observed distances between the centroids of feature i and feature $i+1$. Each l_i is a pair (x_i, y_i) containing the temporal distance and amplitude distance respectively. Let d_i, $1 \leq i \leq k$, be the observed deformation (defined in the next section) between local feature q_i and the observed data at location i in the sequence. Let $D_h = \{d_1, \ldots, d_k, l_1, \ldots, l_{k-1}\}$ be a particular set of observed deformations and distances corresponding to set of candidate features. We will refer to D_h as a *candidate hypothesis.* We can rank candidate hypotheses by evaluating the *likelihood* $p(D_h|Q)$. It remains to define the "generative" probability model $p(D_h|Q)$.

Models for $p(D_h|Q)$ can be defined to varying levels of complexity depending on both (1) the independence structure of the model, and (2) the functional forms of the component probability distributions. In this paper we illustrate the concept with a simple model which assumes feature independence and uses simple parametric distributions. However, in general, the model could incorporate much more complex dependencies such as pattern dependence on global "hidden" scale and deformation variables (see Smyth, Heckerman and Jordan (1996) for a discussion of how to efficiently construct and utilize such models using graph-theoretic formalisms). For our simple model, we have:

$$
\begin{aligned}
& p(D_h|Q) \\
& = p(d_1, \ldots, d_k, l_1, \ldots, l_{k-1}|q_1, \ldots, q_k) \\
& = p(d_k|q_k) \prod_{i=1}^{k-1} p(d_i, l_i|q_i) \\
& \text{(assuming local features} \\
& \text{are generated independently)} \\
& = \prod_{i=1}^{k-1} p(d_i|q_i) \prod_{i=1}^{k-1} p(l_i|q_i) \\
& \text{(assuming deformations and} \\
& \text{distances are conditionally independent).}
\end{aligned}
$$

The models $p(d_i|q_i)$ and $p(l_i|q_i)$ are chosen based on prior knowledge of how the features are expected to be deformed and "spread out." Again we illustrate with some relatively simple models. In this paper we use an exponential model for local deformation distances:

$$p(d_i|Q) = \lambda_i e^{-\lambda_i d_i},$$

which imposes a monotonically decreasing prior belief on deformation distance, i.e., the smaller the deformation, the more likely the observation came from Q. λ_i is chosen to reflect the degree of expected deformation: large λ_i allows less deformation, small λ_i is more permissive.

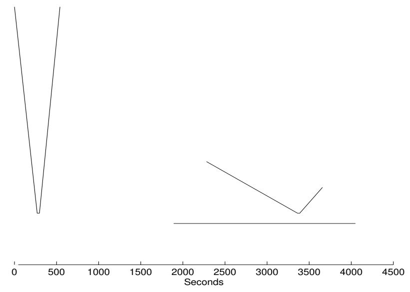

Figure 2: A simple query consisting of 2 feature shapes. The horizontal line at the bottom indicates σ_1, or equivalently, the degree of horizontal elasticity which is allowed between the 2 features.

The inter-feature distance model for $l_i = (x_i, y_i)$ is a joint density on temporal and amplitude elasticity between features. One could for example use bivariate mixture models on x_i and y_i to express complex beliefs on global shape patterns. Here we use a simpler model. We assume that y_i (amplitude) obeys a uniform distribution and is conditionally independent of x_i given q_i. We further assume that x_i obeys a log-normal distribution, or equivalently that $\log x_i$ has a Normal distribution with mean μ_i and variance σ_i^2. μ_i determines how far away features are expected to be and σ_i determines how "elastic" these distances are (e.g., see Figure 2).

Given these particular models, it is straightforward to show that

$$\log p(D_h|Q) \propto \sum_i^k \lambda_i d_i + \frac{1}{2} \sum_i^{k-1} \left(\frac{\log x_i - \mu_i}{\sigma_i} \right)^2,$$

modulo a few extra conditions where this density is zero. Thus, the probabilistic model naturally defines a distance metric which integrates both local and global evidence weighted appropriately according to prior belief. For example, as the λ's are increased, fidelity to local feature shape becomes more important than the distances between features.

Searching for High Likelihood Query Matches

Local Feature Matching

Local feature matching is performed by placing the start of the segmented feature at each breakpoint in the segmented reference sequence and computing the local distance for each location. We use a simple robust method for computing local deformation distances. Consider having placed a feature at a particular reference breakpoint: say there are l breakpoints in the feature and m breakpoints in the part of the reference sequence which does not extend beyond the end of the feature. We vertically project all $l+m$ breakpoints to the "other sequence" to get $l+m$ vertical "projection" distances. The overall deformation distance is defined as the standard deviation of these vertical projection distances. We have found this to be a robust and efficient way to locally match piecewise linear features. The output of this scanning process is a list of roughly K distances, where K is the number of segments in the reference sequence. For a query with Q_f features, this process is repeated for each feature, resulting in a table of size $Q_f \times K$.

Finding Global High-Likelihood Queries

Once we have built the table, we must then search it to find the best possible match for our compound query. The size of the search space scales exponentially with the number of features in the query so we rely on a variety of heuristic search techniques, including greedy ordering and branch-and-bound.

Search Complexity

Let N_R, N_Q, and N_f be the number of data points in the (unsegmented) reference sequence, query subsequence, and feature subsequences, respectively (assume for simplicity that all features have the same number of underlying data points). In a similar manner, let K_R, K_Q, and K_f be the number of segments in the segmented reference sequence, query subsequence, and feature subsequences, respectively. Let $s = N_R/K_R = N_Q/K_Q = N_f/K_f$ be the scaling factor resulting from segmentation (assumed the same across reference, query, and feature sequences for simplicity). Q_f denotes the number of features in query Q (thus, $Q_f = K_Q/K_f = N_Q/N_f$ in this simplified model).

The time complexity of finding the best match for a feature in a reference sequence using "brute force" correlation on the raw data (aka sequential scanning) is $O(N_R N_f)$. The complexity of sequential scanning on segmented data is $O(\frac{N_R N_f}{s^2})$, where $s > 1$, and typically $s >> 1$. Finding the distance tables to set up a query search requires running each of the above searches Q_f times. Exhaustive query search on the distance tables takes $O(N_R^{Q_f})$ and $O((N_R/s)^{Q_f})$ for the unsegmented and segmented data respectively. The application of heuristic search techniques can reduce these times by a factor of $\frac{1}{\alpha}$ where $1-\alpha$ is the fraction of the search space eliminated by the heuristics. Thus, for unsegmented data and brute-force search, the overall time complexity of matching a query scales as $O(N_R N_f + N_R^{Q_f})$, whereas for segmented data with heuristic query search has a time complexity of $O(\frac{N_R N_f}{s^2} + \alpha(N_R/s)^{Q_f})$. For large s and small α the savings are substantial. The experimental results section (below) provides empirical run-time data on real data sets.

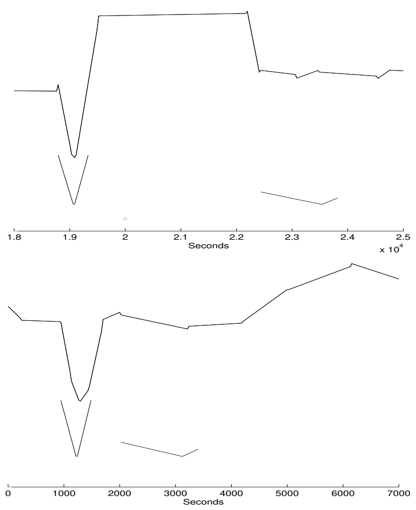

Figure 3: Results of matching the query in Figure 2 with the data in Figure 1, showing the 2 best matches found.

Experimental Results

Due to space limitations we can only present a small subset of our experimental results. Generally, the methods work as one might expect: the matching is relatively insensitive to the exact values of λ and σ and it is quite straightforward to specify query templates and prior distributions. In all of the experiments below, λ_i is set such that the local deformation has a 50% chance of being less than 0.25 $y_i^{\max}$ where $y_i^{\max}$ is the maximum vertical extent of feature i. The μ_i and σ_i parameters are chosen differently for each experiment and the uniform distribution on vertical elasticity between features is made broad enough to essentially make any vertical offsets irrelevant.

As mentioned earlier, the Space Shuttle Mission

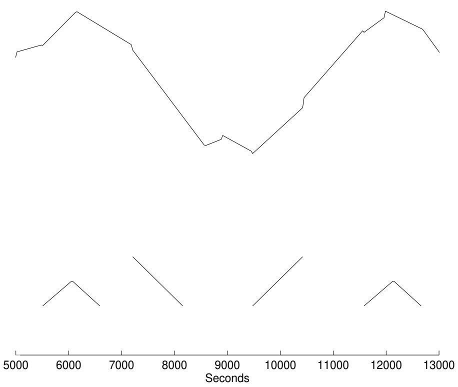

Figure 4: Result of matching a complex query with 4 features on the Shuttle data in Figure 1.

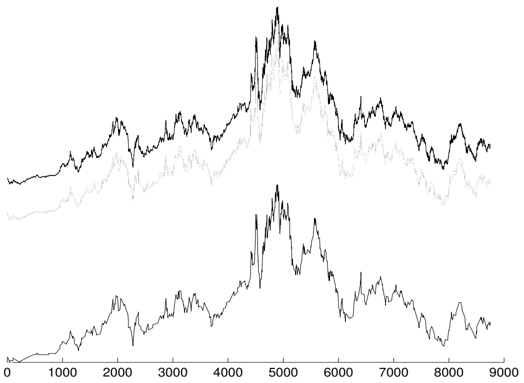

Figure 5: (a) 24 years of daily U.S. 5-Year Treasury Constant Maturity Rate reports, originally 8749 points. (b) segmented into 400 segments using the multi-scale merging algorithm. (Axes in units of days for both figures)

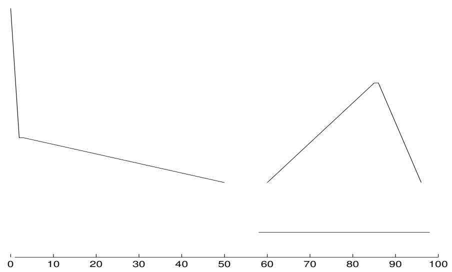

Figure 6: A relatively simple query consisting of 2 feature shapes with σ_1 shown horizontally at the bottom.

Data Archives consist of multiple sensors which are archived once per second for each multi-day shuttle mission. We are investigating the use of fast query matching to support Shuttle mission operations, specifically to facilitate exploration of the vast mission archives for diagnosis, trouble-shooting, and prediction tasks. Figure 2 shows a simple query on the sensor record in Figure 1. The query consists of a steep valley followed by a gentle slope some time later. The mean distance between the two features is 48 minutes. Figure 3 shows the 2 best matches which were found: note that the "elasticity" as encoded by the relatively large value of σ in Figure 2 allows for considerable flexibility in how far away the features are which can be matched. Figure 4 shows the result of matching a more complex query with 2 peaks separated by 2 linear segments.

Another example of the method is provided on the US Daily 5-Year Treasury Constant Maturity Rate. Figure 5 shows the original and segmented data, Figure 6 shows a particular query, a "corner" followed by a peak. Figure 7 shows the three best matches obtained, again showing the flexibility of the approach.

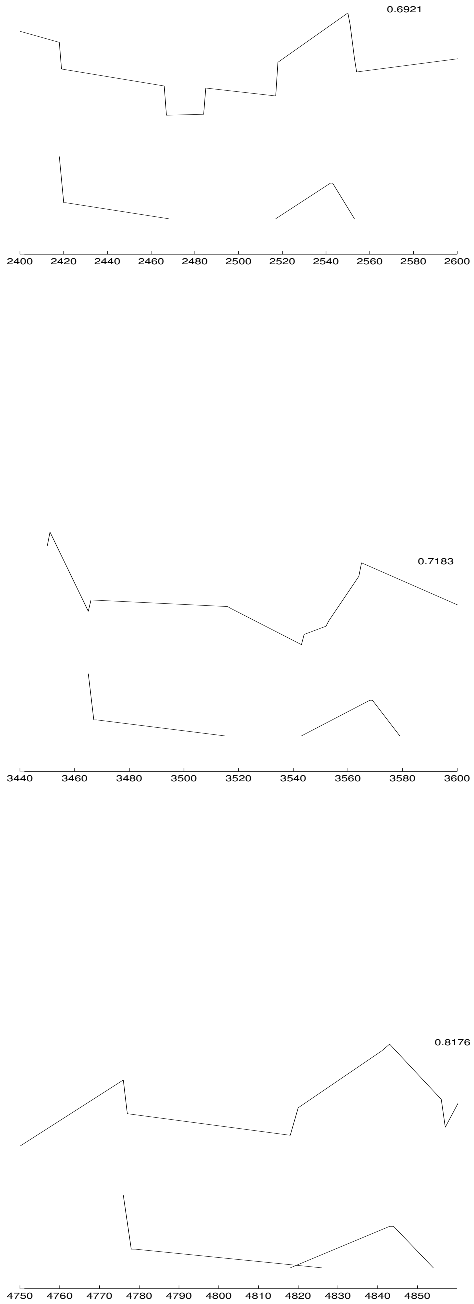

Figure 7: Results of matching the query in Figure 6 with the data in Figure 5, showing the 3 best matches found, with the best at the top. The distance measures are shown alongside, normalized so that a distance of 0 is a perfect match. Axes are in units of days.

Table 1: Experimental and estimated computation times for different search strategies and representations on different sequence data sets and queries. Numbers with asterisks indicate that these quantities were calculated rather than obtained from experimental results. The artificial data were simulated as segmented waveforms, so there is no corresponding raw data set to apply sequential scanning for these sequences.

Name of Data Set	Number of Segments in R	Number of Features in Query Q	Sequential Scanning (seconds)	Segment Matching (seconds)	Exhaustive Table Search (seconds)	Heuristic Table Search (seconds)
Artificial	200	1		2.31	0	0
Artificial	400	1		4.27	0.07	0.06
Artificial	800	1		6.28	0.11	0.11
Artificial	200	2		5.01	148.28	0.22
Artificial	400	2		8.54	598.61	0.27
Artificial	800	2		12.17	2304.55	0.49
Artificial	200	3		7.78	1728*	0.44
Artificial	400	3		13.08	13824*	0.48
Artificial	800	3		19.84	110592*	0.61
Shuttle	43	1	26603	0.92	0.05	0.05
Shuttle	43	2	53200*	1.45	4.94	0.16
Shuttle	43	3	79800*	2.76	9.97	0.39
Treasury	400	1	912	3.98	0.17	0.08
Treasury	400	2	2081	6.78	571.1	0.29
Treasury	400	3	3014	11.22	23000*	0.49

Experimental evaluation of pattern matching systems are somewhat difficult to carry out in a rigorous manner. Researchers tend to use different sequence and query data sets and there is no clear objective "gold standard" for measuring the quality of a particular scheme. An ideal state of affairs would be the widespread use of systematic evaluations on data sets which are common across different studies.

Table 1 summarizes computation times for finding the single best query over a variety of queries and sequences. From the table it is clear that segment matching can be much faster than sequential scanning of the raw data. For complex queries (multiple features), heuristic search universally provides multiple order of magnitude speed-ups over exhaustive search.

Acknowledgments

Part of the research described in this paper was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

References

Agrawal, R., Lin, K. I., Sawhney , H. S and Shim, K. 'Fast Similarity Search in the Presence of Noise, Scaling, and Translation in Time-Series Databases,' In VLDB, pp.490–501, September 1995.

Attneave, F., 'Some informational aspects of visual perception,' *Psychol. Rev.*, 61, 183–193, 1954.

Berndt, D. J., Clifford, J., 'Using Dynamic Time Warping to Find Patterns in Time Series,' in KDD-94: AAAI Workshop on Knowledge Discovery in Databases, 359-370, Seattle, Washington, July 1994.

Burl, M. and Perona, P. 'Recognition of planar object classes,' *Proceedings of the 1996 Computer Vision and Pattern Recognition Conference*, IEEE Press, 1996.

Faloutsos, C., Ranganathan, M., Manolopoulos, Y. 'Fast Subsequence Matching in Time-Series Databases,' *SIGMOD–Proceedings of Annual Conference*, Minneapolis, May 1994.

Pavlidis, T., 'Waveform segmentation through functional approximation,' *IEEE Trans. Comp.*, C-22, no.7, 689–697, 1974.

Pednault, E., 'Minimum length encoding and inductive inference,' in *Knowledge Discovery in Databases*, G. Piatetsky-Shapiro and W. Frawley (eds.), pp.71–92, AAAI Press, 1991.

Shaw, S. W., Defigueiredo, R. J. P., 'Structural Processing of Waveforms as Trees,' *IEEE Trans. ASSP*, Vol.38, No.2, 328–338, February 1990.

Smyth, P., "Hidden Markov models for fault detection in dynamic systems," *Pattern Recognition*, 27(1), 149–164, 1994.

Smyth, P., D. Heckerman, M. Jordan, 'Probabilistic independence networks for hidden Markov probability models,' *Neural Computation*, 9(2), 227–269, 1997.

Wang, J. T., Zhang, K., Jeong, K and Shasha, D. 'A System for Approximate Tree Matching,' *IEEE TKDE.*, 6, no 2 April 1994.