

Markov models for identification of significant episodes

Robert Gwadera *

Purdue University

Department of Computer Sciences

gwadera@cs.purdue.edu

Mikhail Atallah †

Purdue University

Department of Computer Sciences

mja@cs.purdue.edu

Wojciech Szpankowski ‡

Purdue University

Department of Computer Sciences

spa@cs.purdue.edu

Abstract

We propose a new method for a reliable identification of significant sequential episodes occurring *within a window of size w* in an event sequence modeled by a Markov source. As a measure of significance we use $\Omega^{\exists}(n, w)$, the number of windows containing the episode as a subsequence. We prove that $\Omega^{\exists}(n, w)$ is a sum of a φ -mixing sequence of random variables and therefore obeys the central limit theorem. This leads us to a computational formula for a threshold to identify significant episodes. The novelty of our method for Markov source stems from the fact that, instead of scoring the whole sequence using a Markov model, we compute the expected value of $\Omega^{\exists}(n, w)$ and its variance in order to estimate the threshold and compare it to the observed $\Omega^{\exists}(n, w)$. Since performance of the method critically depends on the model structure and parameters, we argue that variable-length Markov models of event streams are superior to fixed-length Markov models. We chose DNA sequences as event sources in experiments, and compared the performance of fixed-length Markov models with interpolated Markov models. This paper is an extension of our previous work in [8, 1] where we considered the problem of the reliable detection of significant episodes for memoryless sources.

*The work of this author was supported by the NSF Grant CCR-0208709, and NIH grant R01 GM068959-01.

†Portions of this author's work were supported by Grants EIA-9903545, IIS-0219560, IIS-0312357, and IIS-0242421 from the National Science Foundation, Contract N00014-02-1-0364 from the Office of Naval Research, by sponsors of the Center for Education and Research in Information Assurance and Security, and by Purdue Discovery Park's e-enterprise Center.

‡The work of this author was supported by the NSF Grant CCR-0208709, NIH grant R01 GM068959-01 and AFOSR Grant FA 8655-04-1-3074.

Keywords: frequent episode mining, probabilistic models

1 Introduction

1.1 Episode mining

Mining episodes was introduced in [11], where the problem of finding frequent episodes in event sequences was defined. An episode was defined as a partially ordered collection of events that occur as a subsequence in a window of a given size in an event stream. The notion of an occurrence is as a subsequence rather than as a substring (that is, contiguity is not required), a requirement dictated by practical considerations because (for example) an “interesting” (e.g., suspicious) sequence of events need not be contiguous in the event stream. An arbitrary episode can be abstractly represented as a directed acyclic graph (DAG), where nodes correspond to events and directed edges define precedence among the events in the episode. Formally, such a graph defines a set of episodes whose members correspond to all distinct paths from the start vertex to end-vertices. We distinguish three types of episodes.

1. A *serial episode* is a sequence of events that occurs in the specified order. In the graph representation a serial episode corresponds to a single path from the first event of the episode to the last one.
2. A *parallel episode* is an unordered collection of events. In the graph representation a parallel episode corresponds to a single node containing all events of the episode. Formally, a parallel episode corresponds to the set of all permutations of symbols of the episode.

3. A *composite episode* corresponds to an arbitrary DAG built from an event and/or an episode by a serial and/or a parallel composition.

In the episode mining we shift the sliding window of a given size w n consecutive events in an event stream T and count the number of windows in which the episode occurred at least once as a subsequence. Note that the same episode may be present in several consecutive windows but within a particular window we count one occurrence even if there are many instances of it in that particular window. Given a window size w and an event sequence T , an episode was defined as *frequent* if its frequency, defined as the fraction of windows in which it occurred at least once, was more than a given frequency threshold τ .

In our work we are interested in episodes that are “significant” (e.g., anomalous); note that the frequency of occurrence is not enough to determine significance (e.g., an infrequent episode might have more significance than a frequent one, depending on the probabilistic characteristics of the event stream).

1.2 Previous work

In our previous papers [8, 1] we assumed that the event sequence T was generated by the Bernoulli (memoryless) source and showed how to compute the threshold τ as well as how to design the window size w such that the discovered frequent episodes are statistically significant. Observe that, for an appropriately large window size any episode will almost surely occur in every window because the probability of its existence in the window of size w $P^\exists(w)$, estimated as a fraction of windows in which the episode occurred, will be close to one. Furthermore, for an appropriately low frequency threshold any episode may be found to be frequent.

Paper [8] considered serial episodes and [1] considered sets of episodes including the special case of the parallel episode. In order to derive the threshold we analyzed $\Omega^\exists(n, w)$, the number of windows of length w , that contain an episode as a subsequence in an event sequence T after n shifts of the sliding window. Using the fact that $\Omega^\exists(n, w)$ is a sum of $w - 1$ dependent random variables we proved that appropriately normalized $\Omega^\exists(n, w)$ is normally distributed, where clearly the expected value $\mathbf{E}[\Omega^\exists(n, w)] = nP^\exists(w)$ and $P^\exists(w)$ is the probability that the episode occurs at least once in a window of length w in an event sequence T over an alphabet \mathcal{A} . We also showed that the variance $\mathbf{Var}[\Omega^\exists(n, w)] \leq cn [P^\exists(w) - (P^\exists(w))^2]$ for $c > 0$. Given the Bernoulli model of an event source, we presented the *upper threshold* for detecting significant *over-*

represented episodes $\tau_u(w) = P^\exists(w) + \frac{b\sqrt{\mathbf{Var}[\Omega^\exists(n, w)]}}{n}$ such that $P\left(\frac{\Omega^\exists(n, w)}{n} > \tau_u(w)\right) \leq \beta(b)$. That is, the

probability that the frequency of the episode $\frac{\Omega^\exists(n, w)}{n}$ is greater than the threshold $\tau_u(w)$ is smaller than $\beta(b)$, i.e., the episode is significant with probabilistic guarantee $1 - \beta(b)$. We also analogously defined the *lower threshold* $\tau_\ell(w)$ for detecting significant *under-*represented episodes such that $P\left(\frac{\Omega^\exists(n, w)}{n} < \tau_\ell(w)\right) \leq$

$\alpha(a)$. The quantity $\frac{\Omega^\exists(n, w)}{n}$ is an estimator of $P^\exists(w)$ denoted $P_e^\exists(w)$. While developing the formula for $P^\exists(w)$ we found a formula for the set of all distinct windows $\mathcal{W}^\exists(w)$ of length w containing the serial episode S of length m at least once as a subsequence. The importance of $\mathcal{W}^\exists(w)$ stems from the fact that $P^\exists(w) = \sum_{x \in \mathcal{W}^\exists(w)} P(x)$ for a Markov model of an arbitrary order including the 0-order (Bernoulli), where $P(x)$ is the probability of x as a string of symbols of length w in a given model. The advantage of the Bernoulli model versus the 1-order Markov or higher is that for the Bernoulli model $P^\exists(w)$ can be computed efficiently exploiting the structure of $\mathcal{W}^\exists(w)$ and the fact that the model requires only $|\mathcal{A}|$ probabilities of symbols of the alphabet \mathcal{A} . Using generating functions and complex asymptotics we presented an asymptotic approximation of $P^\exists(w)$, which is of the form $P^\exists(w) = 1 - \Theta(\rho^w)$ for large w and $0 < \rho < 1$. We provided fast dynamic programming algorithms for computing $P^\exists(w)$ for a serial episode and for an arbitrary set of episodes. In [8] we mined two apparently non-memoryless sources (the English alphabet and the web access data) and showed that, even for these cases, $P^\exists(w)$ closely approximated the actual $P_e^\exists(w)$, which demonstrated that the memoryless assumption did not limit the practical usefulness of the formula. In [1] we mined a large database of *Wal-Mart* transactions for sets of episodes. We also showed that the threshold mechanism indeed provides a sharp detection of significant episodes by continually injecting some episodes until they exceeded the threshold.

1.3 Present work

The present paper extends our previous work to the case of Markov sources that are more applicable and flexible than memoryless sources. The formula for the threshold for Markov models is the same as for the Bernoulli model, the difference is in using conditional probabilities to compute $P^\exists(w)$ and $\mathbf{Var}[\Omega^\exists(n, w)]$. Furthermore, we cannot use the efficient dynamic programming algorithm for computing $P^\exists(w)$ that was designed for the Bernoulli model because in Bernoulli model the probability of a symbol does not depend on its context. There

were three main new challenges that we faced and resolved in this extension. The first was theoretical: in order to use the threshold formula we had to prove that $\Omega^\exists(n, w)$ is a sum of a φ -mixing sequence of random variables meaning that the distant future is practically independent of the present and past and therefore $\Omega^\exists(n, w)$ satisfies the central limit theorem. The second challenge was algorithmic: we had to provide an algorithm for computing $P^\exists(w)$ using conditional probabilities. Finally the third challenge was to select a Markov model structure and a method of parameter estimation to ensure that the prediction of the model is accurate: we suggested variable-length Markov models and in particular, in experiments conducted on DNA sequences, we focused on the interpolated Markov model.

Given an event sequence T , we need to choose an optimal Markov model for the data and given that model we need to choose an optimal method for parameter estimation for our method for detecting significant episodes. Higher order models describe data more accurately but increase the number of excessive parameters. Using a fixed-length Markov model of order k with $|\mathcal{A}|^{k+1}$ parameters can be inefficient since for real-life data the actual memory length varies. The number of model parameters can be significantly reduced by merging equivalent states (contexts of length k) that have identical conditional probabilities. Such reduced models, first considered in [15] were termed the *variable-length Markov chains/models* or *tree models* [20, 12] since they can be conveniently represented with a tree structure. Thus, the advantage of variable-length Markov models over fixed-length models is that they efficiently capture the redundancies that are typical for real-life data. Because of that fact they are particularly well suited for our method for detecting significant episodes since we can efficiently build such models based on our empirical, expert, knowledge of the source of events. For example: sometimes we know that some symbols occur only in n -grams (strings of n ordered elements) and using a full fixed-length Markov model is too excessive and may drastically limit the usefulness of our method while the Bernoulli model is too inaccurate to capture the dependency of symbols in the n -grams.

Formally, given an alphabet $\mathcal{A} = \{a_1, a_2, \dots, a_{|\mathcal{A}|}\}$ a k -order variable-length Markov model can be represented as a *context tree* [15]. Let $\mathcal{C} = \{c_1, c_2, \dots, c_{|\mathcal{S}|}\}$ be the set of contexts in a k -order variable-length Markov model where $c_j = c_j[|c_j|] \dots, c_j[2], c_j[1]$ is the j -th context of length $1 \leq |c_j| \leq k$, written as a time-reversed string, where $c_j \in \mathcal{A}^{|c_j|}$. A context tree is a suffix tree, built from the contexts in \mathcal{C} that is called a *model*. The set of parameters of a k -order variable-length Markov model is defined as $\Theta = \{\theta_{1,1}, \theta_{1,2}, \dots, \theta_{|\mathcal{A}|,|\mathcal{C}|}\}$ where

$\theta_{i,j} = P(a_i|c_j)$ is the conditional probability of generating the symbol a_i given the context c_j subject to $\sum_{i=1}^{|\mathcal{A}|} P(a_i|c_j) = 1$. Thus, \mathcal{C} is the set of states and Θ is the set of transition probabilities in a k -order variable-length Markov chain. In [15] the *context* algorithm was presented for estimation of the minimal state space and the parameters of a variable-length Markov model. In [16] minimization of stochastic complexity of a source in a given model was suggested as a criterion for selecting an optimal (minimal) model. In [21] the *context tree weighting* algorithm was proposed for computing probability $P(T)$ of a Markov source T using an average over all possible models having orders less than a given order k . However the optimal model selection problem for the class of variable-length Markov models is still not well understood. Furthermore the parameter estimation from sparse data presents another problem.

Therefore we turn our attention to a class of variable-length Markov models called interpolated Markov model (IMM). IMM does not optimize the state space but builds a variable-length Markov model implicitly as a result of parameter estimation from sparse data. Given an alphabet of size $|\mathcal{A}|$, we could model the event stream T by a k -order fixed-length Markov model with \mathcal{A}^{k+1} conditional probabilities to be estimated from the training data. If we are not interested in optimality of the model then a higher-order model should always do at least as well as, and frequently better than a lower-order model. In practice, when using a k -order model, if the training sequence is of length N then there are only $N - k$ strings of size $k + 1$ available to estimate $|\mathcal{A}|^{k+1}$ conditional probabilities and frequencies of some of the $|\mathcal{A}|^k$ context strings become too small or even zero. Deriving a model of too high order from such sparse data will lead to over-fitting. More formally, let $c_j = c_j[k] \dots, c_j[2], c_j[1]$ be the j -th context of length k , written as a time-reversed string, in a k -order fixed-length Markov model, where $c_i \in \mathcal{A}^k$. We estimate the conditional probabilities $P(a_i|c_j)$ using the maximum likelihood (ML) estimate given by the following formula $P(a_i|c_j) = \frac{n(c_j a_i)}{\sum_{i=1}^{|\mathcal{A}|} n(c_j a_i)}$, where $n(c_j a_i)$ is the frequency of the string $c_j a_i$ observed in the training set. Notice that even if the training set is too short to accurately estimate all probabilities, for some probabilities $P(a_i|c_j)$ the number of occurrences of the string $c_j a_i$ will be sufficient and should be accepted by the model. The problem of parameter estimation of Markov models from sparse data is known as *smoothing* and has been widely discussed in the literature on language modeling [6]. The smoothing is a technique for adjusting the maximum likelihood estimates of probabilities to produce more accurate probabilities. The

name smoothing comes from the fact that these methods tend to make the probabilities more uniform, by adjusting low probabilities upward and higher probabilities downward. Not only do smoothing methods generally prevent zero probabilities, but they also improve the accuracy of the model as the whole. Whenever a probability is estimated from a fewer counts, smoothing has the potential to significantly improve estimation.

Techniques as *back-off* [10] and *interpolation* [13] have been implemented to deal with sparse data. The back-off model backs off to lower order models depending on counts of respective contexts. The interpolated model is a Markov chain with a new structure, where a conditional probability of order k is a combination of equal and lower order probabilities weighted by interpolation parameters, giving high weight to probability estimates corresponding to high frequency contexts and lower weight to estimates corresponding to low frequency contexts. A further extension of IMM is *interpolated context model* (ICM) [7]. While in IMM we estimate the probability $P(a_i|c_j)$ of a symbol a_i based on variable length contexts immediately preceding a_i the ICM is more flexible and general by allowing to choose any contexts (not just those adjacent to a_i).

In this paper we focus on the interpolated Markov model and we compare its performance with fixed-length Markov models for detecting significant episodes. We use the notation $c_j[1 : n]$ for $n = k, k - 1, \dots, 1$ to denote a suffix of length n of context c_j of length k and we omit the notation for $n = k$, i.e., we write c_j instead of $c_j[1 : k]$ in cases where k is implied. We are interested in Markov models that define conditional probabilities $P(a_i|c_j)$ as a linear combination of conditional probabilities corresponding to suffixes of c_j . The following recursion defines a value of the interpolated conditional probability in IMM:

$$P_{IMM}(a_i|c_j) = \lambda(c_j) \cdot P(a_i|c_j) + (1 - \lambda(c_j)) \cdot P_{IMM}(a_i|c_j[1 : k - 1]),$$

where $0 \leq \lambda(c_j) \leq 1$ and $P(a_i|c_j)$ is the probability estimate using the maximum likelihood (ML) estimate from the training data. For contexts c_j not observed in the training data, i.e., if $n(c_j a_i) = 0$ then we set $P(a_i|c_j) = P(a_i|c_j[1 : n])$ for $n = \max_{1 \leq n \leq k} \{n | n(c_j[1 : n], a_i) > 0\}$ and this is exactly the place in the computation of parameters of an IMM where a variable-length Model is being implicitly built. The value of the parameter $\lambda(c_j)$ can be interpreted in many ways. Following [17] interpretation of the parameter depends on the following interpretations of the IMM:

- Context model interpretation: the parameters combine the predictions from contexts of varying

length. Since longer contexts support stronger predictions and shorter contexts have more accurate statistics the interpolation of the predictions of different context lengths results in more accurate prediction than from a fixed context.

- State model interpretation: the parameters are hidden transitions from a higher order Markov model to a lower Markov model where the interpolation parameters model our beliefs about how much of the past is necessary to predict a state transition in an underlying Markov source of unknown order.
- Nonuniform model interpretation: the parameters are beliefs about conditional independence with probability $(1 - \lambda(c_j))$ that the future does not depend on $c_j[k]$.

In general if the frequency of context c_j is sufficiently high, the value of $\lambda(c_j)$ is close to 1. In the opposite case $\lambda(c_j)$ is close to zero and the interpolation probability $P_{IMM}(a_i|c_j)$ gains more from $P_{IMM}(a_i|c_j[1 : k - 1])$. However the problem of finding interpolation parameters is still more of an art than an exact science. In our experiments we assumed a given order of IMM and used a modification of the method based on χ^2 -test introduced in [18].

We conduct our experiments on genomic data represented as strings of nucleotide symbols over the alphabet $\mathcal{A} = \{A, C, G, T\}$. Markov models of DNA sequences have frequently been used in gene finding algorithms [18], where the interest was in finding strings of symbols instead of subsequences in the form of episodes. The novelty of our approach is that we treat the genomic sequence as a stream of symbols generated by a Markov model of an unknown order and we do not consider any biological structures as coding/non-coding regions in the DNA. Furthermore we do not score the testing sequence using a trained Markov model, as in the work on gene discovery, in order to determine whether the sequence has been generated by the model. Adapting the method of scoring the sequence for episode discovery would mean training a separate Markov model for every combination of window length and episodes type. Note that in the episode framework we consider $\Omega^{\exists}(n, w)$ the number of windows containing an episode as a subsequence, which is a function of a Markov chain rather than a well defined structure (coding/non-coding regions) of the sequence as in the gene discovery methods. Therefore in our method for the reliable detection of significant episodes we use a Markov model only to compute the expected value and variance of $\Omega^{\exists}(n, w)$ needed for the threshold computation. Because of the sequential nature of Markov sources we consider only serial episodes while using Markov models. We do not test

the threshold $\tau_u(w)$ directly in experiments by computing its value and simulating occurrences of significant episode as in [8, 1] because we already showed in [8, 1] that the accuracy of the threshold is determined by prediction accuracy of the formula for $P^\exists(w)$. Therefore we test the threshold indirectly by focusing on the predictive performance of the formula for $P^\exists(w)$ for Markov models.

Perhaps the most intriguing question is whether we can improve our detection method on DNA data in terms of accuracy by employing a Markov model rather than the Bernoulli model. As we will see in experiments the answer to this question is affirmative.

The paper is organized as follows. Section 2 presents example applications of our theory. Section 3 presents our main results containing theoretical foundation. Section 4 contains experimental results demonstrating the applicability of the derived formulas.

2 Applications of our method

There are multiple uses for the theory we developed. Given a probabilistic model of an event sequence T , example applications of our method include:

- *Designing the sliding window size:* given a priori knowledge of episodes of interest, we can select an appropriate window size w such that the discovered episodes are meaningful.
- *Validation of the sliding window size:* given a window size w and an episode (e.g., a frequent episode) discovered in the event stream T , we can validate the window size w for the discovered episode.
- *Identification of significant episodes:* given a window size w and an episode (e.g., a frequent episode) discovered in the event stream T , we can determine whether the episode is significant.
- *Episode ranking:* given a collection of episodes (e.g., all frequent episodes) discovered in the event stream T , we can rank the episodes with respect to their significance.

Given a probabilistic model of an event sequence T and an episode with observed frequency $\frac{\Omega^\exists(n,w)}{n}$, the episode can be classified using the upper threshold $\tau_u(w)$ and the lower threshold $\tau_\ell(w)$ as follows:

- *significant:*
 - if $\frac{\Omega^\exists(n,w)}{n} > \tau_u(w)$ for over-represented episodes

- if $\frac{\Omega^\exists(n,w)}{n} < \tau_\ell(w)$ for under-represented episodes

- *normal:* if $\frac{\Omega^\exists(n,w)}{n} \in [\tau_\ell(w), \tau_u(w)]$
- *meaningless:* if $\frac{\Omega^\exists(n,w)}{n} \approx 1$ and $P^\exists(w) \approx 1$ meaning that the window size w is too large.

3 Analytical results

3.1 Definition of the problem of identification of significant episodes

For clarity of the presentation we analyze only the case of a single serial episode $S = S[1]S[2] \dots S[m]$ of length m but the results can be generalized to an arbitrary set of serial episodes. Of course we do not analyze parallel episodes since they are unordered sequences.

The problem of identification of significant episodes in a Markov source T can be stated as follows.

Given:

- an alphabet $\mathcal{A} = \{a_1, a_2, \dots, a_{|\mathcal{A}|}\}$
- a k -order variable-length Markov model of the source T with parameters represented as follows:
 - $\mathcal{C} = \{c_1, c_2, \dots, c_{|S|}\}$ is the set of contexts where $c_j = c_j[|c_j|] \dots c_j[2], c_j[1]$ is the j -th context of length $1 \leq |c_j| \leq k$, written as a time-reversed string and $c_j \in \mathcal{A}^{|c_j|}$.
 - $\Theta = \{\theta_{1,1}, \theta_{1,2}, \dots, \theta_{|\mathcal{A}|,|c_j|}\}$ is the set of parameters where $\theta_{i,j} = P(a_i|c_j)$ is the conditional probability of generating the symbol a_i given the context c_j subject to $\sum_{i=1}^{|\mathcal{A}|} P(a_i|c_j) = 1$.
- $\Omega^\exists(n,w)$, the observed number of windows of length w containing at least one occurrence of a serial episode $S = S[1]S[2] \dots S[m]$ after n shifts of the window
- a level $\beta(b)$ (e.g., $\beta(b) = 10^{-5}$),

is the observed episode S significant?

In Section 3.2 we prove that $\Omega^\exists(n,w)$ is normally distributed (Theorem 1). This will allow us to compute the threshold as follows

$$(1) \quad \begin{cases} \tau_u(w) &= P^\exists(w) + \frac{b\sqrt{\mathbf{Var}[\Omega^\exists(n,w)]}}{n} \\ \beta(b) &= \frac{1}{\sqrt{2\pi}} \int_b^\infty e^{-\frac{t^2}{2}} dt \end{cases}$$

where $\mathbf{Var}[\Omega^\exists(n,w)] \leq [n + (2n - w)(w - 1)][P^\exists(w) - (P^\exists(w))^2]$.

Thus, if $\frac{\Omega^\Xi(n,w)}{n} > \tau_u(w)$ then episode S is significant with probabilistic guarantee $1 - \beta(b)$, i.e., $P\left(\frac{\Omega^\Xi(n,w)}{n} > \tau_u(w)\right) \leq \beta(b)$.

In section 3.3 we provide an algorithm for computing $P^\Xi(w)$.

3.2 Central limit for $\Omega^\Xi(n, w)$

In this section we show that $\Omega^\Xi(n, w)$ is a sum of φ_n -mixing sequence of random variables and therefore it satisfies the central limit theorem even though independence of the random variables summing to $\Omega^\Xi(n, w)$ is clearly violated.

We consider a stationary and ergodic infinite k -order Markov source T .

Definition 1 A k -order Markov source is a sequence of random variables t_1, t_2, \dots over an alphabet $\mathcal{A} = \{a_1, a_2, \dots, a_{|\mathcal{A}|}\}$ with the Markov property

$$P(t_1, \dots, t_n) = P(t_1, \dots, t_k) \cdot \prod_{i=k+1}^n P(t_i | t_{i-1}, \dots, t_{i-k})$$

where k is the minimum integer for which the Markov property holds.

A k -order fixed-length Markov source can be described by a finite state Markov chain.

Definition 2 A finite state k -order Markov chain is a sequence of random variables Q_1, Q_2, \dots , where $Q_i = (t_{i-1}, t_{i-2}, \dots, t_{i-k})$ is a symbol from a finite state alphabet \mathcal{Q} of cardinality $|\mathcal{Q}| = |\mathcal{A}^k|$ and there is a transition from state $Q_i = (t_{i-1}, t_{i-2}, \dots, t_{i-k})$ to state $Q_j = (t_i, t_{i-1}, \dots, t_{i-k+1})$ with transition probability $P(t_i | t_{i-1}, \dots, t_{i-k})$.

We use the Markov chain model of a k -order Markov source in the proof that $\Omega^\Xi(n, w)$ is a sequence of φ -mixing sequence of random variables.

A formal definition of a φ -mixing sequence is as follows.

Definition 3 Let $\varphi_1, \varphi_2, \dots$ be a sequence of numbers such that $\varphi_n \rightarrow 0$. A stationary sequence of random variables X_1, X_2, \dots, X_n is φ -mixing if $|P(E_2 | E_1) - P(E_2)| \leq \varphi_n$ for every j and $E_1 \in (X_1, \dots, X_j)$ and every k and $E_2 \in (X_{j+n}, \dots, X_{j+n+k})$.

In this definition, E_1 is an event that depends only on X_1, \dots, X_j , and E_2 is an event that depends only on $X_{j+n}, \dots, X_{j+n+k}$. The condition requires that E_1 and E_2 are almost independent in the sense that $|P(E_2 | E_1) - P(E_2)|$ is small for large n .

Now we derive the normal limiting distribution of $\Omega^\Xi(n, w)$. Observe that

$$\Omega^\Xi(n, w) = \sum_{i=1}^n I_i^\Xi(w)$$

where

$$I_i^\Xi(w) = \begin{cases} 1 & \text{the episode } S \text{ occurs at least once as a} \\ & \text{subsequence in the window ending at} \\ & \text{position } i \text{ in } T; \\ 0 & \text{otherwise,} \end{cases}$$

where i is the relative position with respect to the first position ($i = 1$). Thus, we easily have $\mathbf{E}[I_i^\Xi(w)] = P^\Xi(w)$, $\mathbf{Var}[I_i^\Xi(w)] = P^\Xi(w) - (P^\Xi(w))^2$ and $\mathbf{E}[\Omega^\Xi(n, w)] = nP^\Xi(w)$.

Thus, the independence of the sequence of $I_i^\Xi(w)$ for $1 \leq i \leq n$ is violated twofold since:

1. observation windows overlap within $w - 1$ events meaning $|P(I_{i+k}^\Xi(w) = 1 | I_i^\Xi(w) = 1) - P(I_{i+k}^\Xi(w) = 1)| \neq 0$ for $1 \leq k \leq w - 1$
2. the event sequence T is not memoryless meaning $|P(I_{i+k}^\Xi(w) = 1 | I_i^\Xi(w) = 1) - P(I_{i+k}^\Xi(w) = 1)| \neq 0$ for $k > w - 1$.

For Markov sources the central limit theorem holds as long as $|P(I_{i+k}^\Xi(w) = 1 | I_i^\Xi(w) = 1) - P(I_{i+k}^\Xi(w) = 1)| \rightarrow 0$ as $k \rightarrow \infty$, i.e., I_{i+k}^Ξ and I_i^Ξ are practically independent as k becomes large meaning the sequence $I_1^\Xi(w), I_2^\Xi(w), \dots, I_n^\Xi(w)$ is φ -mixing.

Notice that according to the definition of a φ -mixing sequence the $w - 1$ -dependent sequence $I_1^\Xi(w), I_2^\Xi(w), \dots, I_n^\Xi(w)$ is φ -mixing with $\varphi_n = 0$ for $|i - j| > w - 1$.

Given a k -order Markov event source T , $I_i^\Xi(w)$ is a function of the corresponding Markov chain of order k . Therefore to prove that $I_1^\Xi(w), I_2^\Xi(w), \dots, I_n^\Xi(w)$ is φ -mixing we use the fact that the Markov chain is φ -mixing. According to [3], let Y_1, Y_2, \dots, Y_n be a Markov chain with finite state space and positive transition probabilities p_{ij} . Let $X_i = f(Y_i)$, where f is some real function of the state space. If the initial probabilities are stationary then Y_1, Y_2, \dots, Y_n is stationary. Moreover $|p_{ij}^n - p_j| \leq \rho^n$ where $\rho < 1$. Let r be the number of states. Then according to [3] Example 27.6 X_1, X_2, \dots, X_n is φ_n -mixing with $\varphi_n = r\rho^n$.

Based on Theorem 27.5 in [3] and the fact that $I_1^\Xi(w), I_2^\Xi(w), \dots, I_n^\Xi(w)$ is φ -mixing the central limit theorem holds for $\Omega^\Xi(n, w)$ in a k -order Markov source.

Theorem 1 The random variable $\Omega^\Xi(n, w)$ obeys the Central Limit Theorem in the sense that its distribution

is asymptotically normal, for $a, b = O(1)$ we have

$$\lim_{n \rightarrow \infty} P \left\{ a \leq \frac{\Omega^\exists(n, w) - \mathbf{E}[\Omega^\exists(n, w)]}{\sqrt{\mathbf{Var}[\Omega^\exists(n, w)]}} \leq b \right\} = \frac{1}{\sqrt{2\pi}} \int_a^b e^{-\frac{t^2}{2}} dt$$

for fixed w .

Theorem 1 leads to the Formula (1) for the threshold.

3.3 Algorithm for computing $P^\exists(w)$

In this section we present an algorithm for computing $P^\exists(w)$ for Markov models.

The probability of existence of an episode can be expressed as follows

$$(2) \quad P^\exists(w) = \sum_{x \in \mathcal{W}^\exists(w)} P(x).$$

where $\mathcal{W}^\exists(w)$ is the set of all distinct windows of length w containing the episode as a subsequence. Let x_i be the i -th symbol of a window $x \in \mathcal{W}^\exists(w)$ then the probability of the window can be computed as follows

$$P(x) = P(x_0)P(x_1|x_0) \dots P(x_{k-1}|x_{k-2} \dots x_0) \cdot \prod_{i=k}^{w-1} P(x_i|x_{i-1} \dots x_{i-k})$$

where k is the order of the model. In our papers [8, 1] we gave formulas for $\mathcal{W}^\exists(w)$ for a single serial episode $S = S[1]S[2] \dots S[m]$ and for an arbitrary set of serial episodes $\mathbf{S} = \{S_1, S_2, \dots, S_m\}$ respectively. In particular, $\mathcal{W}^\exists(w)$ for a single serial episode can be enumerated as follows

$$\mathcal{W}^\exists(w) = \bigcup_{\sum_{k=1}^{m+1} n_k = w-m} \overline{S[1]}^{n_1} \times S[1] \times \dots \times \overline{S[m]}^{n_m} \times S[m] \times \mathcal{A}^{n_{m+1}},$$

where \bar{a} denotes $\mathcal{A} - a$ for $a \in \mathcal{A}$. Using formula (3) directly is computationally very expensive. Furthermore computing $P(x)$ for every window x independently would be inefficient because many windows share the same prefix. Therefore we propose a computational method where we enumerate the windows according to the depth-first traversal of a trie build from the members of $\mathcal{W}^\exists(w)$ without the trailing $\mathcal{A}^{n_{m+1}}$ that contributes a factor of 1 to the computation of the probability. The idea of this method is that the probability of each distinct prefix of the set of windows $\mathcal{W}^\exists(w)$ is computed once. An example of such a trie for $S = abc$ and $w = 4$ is shown in Figure 1.

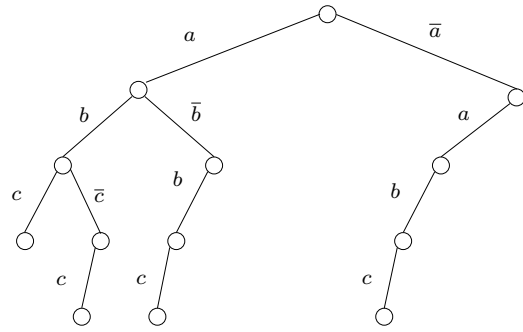


Figure 1: A trie for $\mathcal{W}^\exists(3)$ the set of windows of length $w = 4$ containing $S = abc$ as a subsequence

4 Experimental results

The ultimate measure of a statistical model is its predictive performance in the domain of interest. Therefore in experiments we compare the actual (observed) $P_e^\exists(w) = \frac{\Omega^\exists(n, w)}{n}$ value with $P^\exists(w)$ obtained using a trained model. As we stated in the introduction, we consider only a serial episode S of length m in experiments. We used an algorithm for finding occurrences of a serial episode to find $\Omega^\exists(n, w)$. To evaluate the performance of a model we used the following distance measure between two functions

$$(3) \quad d = \left[\frac{1}{r} \sum_{i=1}^r \frac{|P_e^\exists(w_i) - P^\exists(w_i)|}{P_e^\exists(w_i)} \right] 100\%$$

where $w_1 < w_2 < \dots < w_r$ are the tested window sizes. We tested the prediction of Markov models on the following genomic sequences: *Haemophilus influenzae* of length 1,830,025, *Helicobacter pylori* of length 1,667,826 and *Human chromosome 22* two segments of length 234,227 and 3,661,561 respectively. We estimated the conditional probabilities using the maximum likelihood estimator for both the fixed-length and interpolated model IMM. We used *Helicobacter pylori* and the first segment of *Human chromosome 22* as training sets. For each training set we built a k -order fixed-length models and a k -order IMM for $k = 0, 1, 2, 3, 4, 5$. All our algorithms have been implemented in C++ and run under Linux operating system. The IMM algorithm is presented in Section 4.1.

4.1 Interpolated Markov model

We used a modification of the χ^2 -confidence based interpolation method introduced in the GLIMMER gene finding algorithm in [18] for computing the $\lambda(c_i)$ in the equation for the conditional probability in the

interpolated Markov model

$$P_{IMM}(a_i|c_j) = \lambda(c_j) \cdot P(a_i|c_j) + (1 - \lambda(c_j)) \cdot P_{IMM}(a_i|c_j[1 : k-1]).$$

Algorithm 1: k -order IMM parameter estimation

input : $n(c_j), n(c_j, a_i), N, k$
output: k -order $P_{IMM}(a_i|c_j)$

begin

for $j = 1$ **to** $|\mathcal{A}|^k$ **do**

$th = (N - k + 1)P(c_j)$;

if $n(c_j) \geq th$ **then**

$\lambda(c_j) = 1$

else

$chisquare = 0$;

for $i = 1$ **to** $|\mathcal{A}|$ **do**

$chisquare += \frac{(n(c_j, a_i) - n(c_j) \cdot P_{IMM}(a_i|c_j[1:k-1]))^2}{n(c_j) \cdot P_{IMM}(a_i|c_j[1:k-1])}$;

$p = \text{gammp}(chisquare, \mathcal{A} - 1)$;

if $p < 0.5$ **then**

$\lambda(c_j) = 0$

else

$\lambda(c_j) = \frac{p \cdot n(c_j)}{th}$

for $i = 1$ **to** $|\mathcal{A}|$ **do**

$P_{IMM}(a_i|c_j) = \lambda(c_j) \frac{n(c_j, a_i)}{n(c_j)} + (1 - \lambda(c_j)) \cdot P_{IMM}(a_i|c_j[1 : k-1])$

end

end

The algorithm takes as its input the following parameters: $n(c_j)$ the frequency of context c_j , $n(c_j, a_i)$ the frequency of string $c_j a_i$, k the order of the IMM and N the length of the training set. The function $\text{gammp}(chi, df)$ computes the probability that the χ^2 random variable is smaller than $chisquare$ i.e. it computes the cumulative distribution function of the χ^2 for $\mathcal{A} - 1$ degrees of freedom. The GLIMMER system used a fixed value for $th = 400$. We interpreted the threshold as the expected number of occurrences of a context c_i of length i as a string in the training set for 0-order Markov source. Thus, we set $th = \mathbf{E}[n(c_j)] = (N - k + 1)P(c_j)$, where $P(c_j)$ is the probability of the context in the 0-order Markov model. Alternatively we could use $th = (N - k + 1)P(c_j) - \sqrt{\mathbf{Var}[n(c_j)]}$. The following sections use the IMM computed by Algorithm 1.

4.2 Fixed-length versus IMM for the same training and testing source

In this experiment we experimentally confirmed the correctness of our theoretical results including the proof of the central limit theorem, the derived formula for $P^\exists(w)$ and the algorithm for computing $P^\exists(w)$. We expected to achieve a better prediction accuracy using the 5-order (fixed-length and IMM) comparing to the 0-order. To exclude a possibility of model misbehavior (over-fitting, etc.) we used the the same sequence of *Haemophilus influenzae* as a training and testing source. We set a serial episode $S = CCGT$ and for each k -order fixed-length model for $k = 0, 1, 2, 3, 4, 5$ and 5-order IMM we computed $P^\exists(w)$ and compared to an observed $P_e^\exists(w)$ for $w = [5, 20]$ by computing the prediction error using Equation 3. The computed prediction errors are represented by a bar graph in Figure 2. Clearly the prediction error decreases monotonically starting from 1-order fixed-length model up. 5-order (fixed-length and IMM) gives the best prediction significantly outperforming the 0-order. This validates our theoretical and algorithmic results. 5-order IMM performs closely to 5-order fixed-length model since the training source was sufficiently large and the IMM did not use the lower order models.

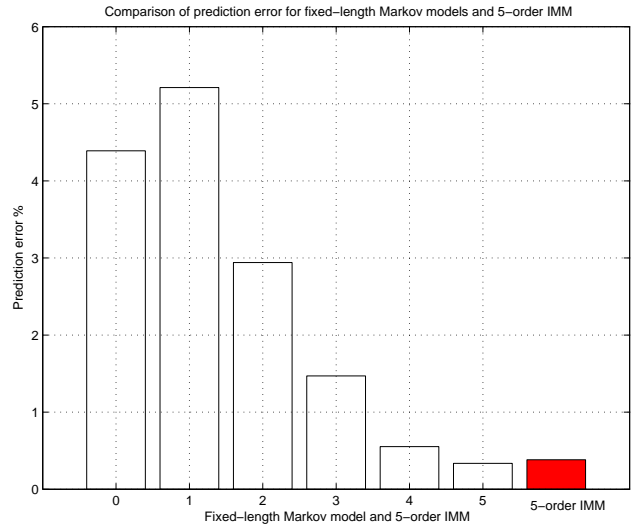


Figure 2: Prediction error d between $P^\exists(w)$ (computed) and $P_e^\exists(w)$ (observed) for a serial episode using a k -order fixed-length Markov models for $k = 0, 1, 2, 3, 4, 5$ and 5-order IMM

4.3 Fixed-length versus IMM for the same training source and a different testing source

In this experiment we compared the fixed-length 5-order with 5-order IMM. We used *Haemophilus influenzae* for computing the conditional probabilities and we tested the performance of both models on *Helicobacter pylori*. We expected the IMM to perform better than the fixed-length model because of its smoothing properties while we expected the fixed-length model to suffer from over-fitting. Also we did not expect a significant improvement in accuracy of IMM because the training set of size (1,830,025) was sufficiently large to find all context strings. We set a serial episode $S = CCGT$ and for each k -order fixed-length model for $k = 0, 1, 2, 3, 4, 5$ and 5-order IMM we compared $P^\exists(w)$ with the observed $P_e^\exists(w)$ for $w = [5, 20]$ by computing the prediction error given in Equation 3. The results, shown as a bar graph in Figure 3 confirm our expectations and the IMM performed slightly better than 5-order fixed-length model. Also 1-order fixed-length turned out to be the winner probably because there is a difference in the structure of DNA of *Helicobacter pylori* and *Haemophilus influenzae* and the 1-order captured the necessary structure without over-fitting.

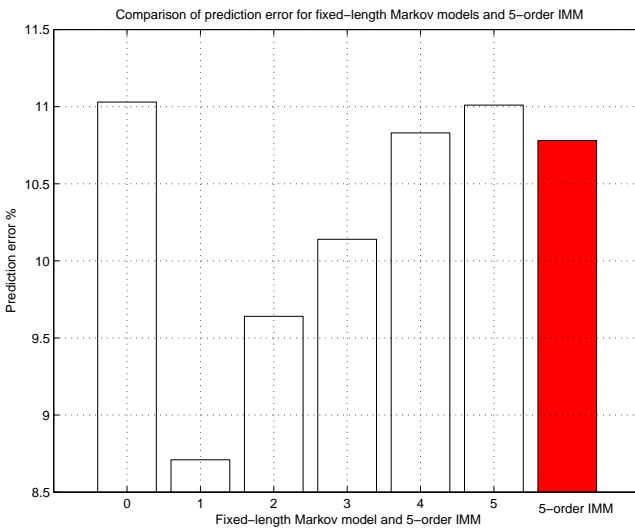


Figure 3: Prediction error d between $P^\exists(w)$ (computed) and $P_e^\exists(w)$ (observed) for a serial episode using a k -order fixed-length Markov models for $k = 0, 1, 2, 3, 4, 5$ and 5-order IMM

4.4 Fixed-length versus IMM for sparse data

In this experiment we wanted to check whether the 5-order IMM outperforms the 5-order fixed-length model when both are trained from sparse training data. To accomplish it we chose the first segment of *Human chromosome 22* of length 234,227 as a training set and we tested both models on the second segment of the same chromosome of length 3,661,561. We set a serial episode $S = CCGT$ and computed $P^\exists(w)$ for both models and compared to the observed $P_e^\exists(w)$ for $w = [5, 20]$. We plotted the results in Figure 4 to show the shape of the curves. From the figure we can see that 5-order IMM slightly better approximates the observed $P_e^\exists(w)$.

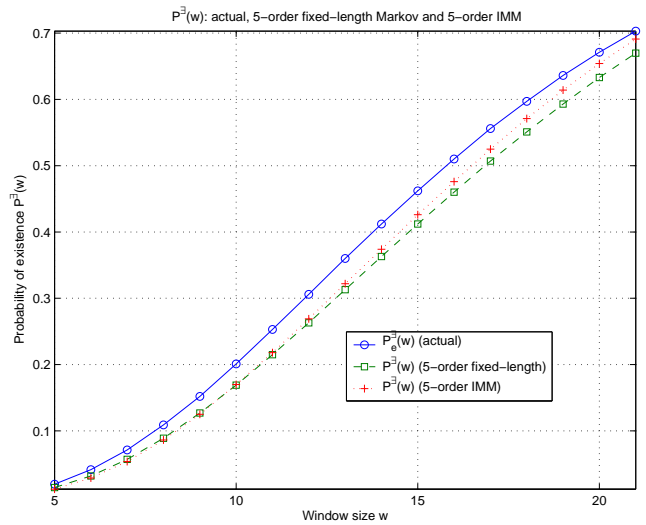


Figure 4: Observed $P_e^\exists(w)$ and computed $P^\exists(w)$ for a serial episode using 5-order fixed-length Markov model and 5-order IMM

5 Conclusions and extensions

We presented a new method for a reliable identification of significant episodes in variable-length Markov source. As a measure of significance we used $\Omega^\exists(n, w)$ the number of windows in which the episode occurred in the event stream. We proved that $\Omega^\exists(n, w)$ is a sum of so called φ -mixing random variables and obeys the central limit theorem, which leads to a computational formula for the threshold $\tau_u(w)$ for discovering significant episodes. We proposed to use variable-length Markov models with the threshold mechanism because of their flexibility for modeling a wide variety

of event sources. In particular we compared the interpolated Markov model with the fixed-length Markov model in experiments conducted on DNA sequences. We showed that the IMM slightly outperforms the fixed-length model in terms of prediction accuracy. We also showed that for DNA source the use of Markov models outperforms memoryless models in terms of accuracy in predicting occurrences of episodes even though a Markov model can be susceptible to over-fitting. The drawback of using Markov models is the high computational cost of computing the threshold. This could be overcome by using a combination of a Bernoulli model and a Markov model. In such a technique we could use a Markov model for small values of w where the accuracy of the prediction would be crucial and we could use the Bernoulli model for large w , where $P^{\exists}(w)$ for both a Markov and the Bernoulli model converge to 1.

References

- [1] M. Atallah, R. Gwadera and W. Szpankowski, Detection of significant sets of episodes in event sequences, *Fourth IEEE International Conference on Data Mining*, pages 67-74, Brighton UK.
- [2] R. Azad and M. Borodovsky, Effects of choice of DNA sequence model structure on gene identification accuracy, *Bioinformatics* 2004.
- [3] P. Billingsley (1986), *Probability and measure*, John Wiley, New York.
- [4] L. Boasson, P. Sequels, I. Guessarian, and Y. Matiyasevich (1999), Window-Accumulated Subsequence Matching Problem is Linear, *Proc. PODS*, 327-336.
- [5] S. Brin, R. Motwani, C. Silverstein, *Beyond Market Baskets: Generalizing Association Rules to Correlations*, SIGMOD 1997, Proceedings ACM SIGMOD International Conference on Management of Data, May 13-15, 1997, Tucson, Arizona
- [6] S. Chen, J. Goodman, An Empirical Study of Smoothing Techniques for Language Modeling, Technical Report TR-10-98, Computer Science Group, Harvard University, 1998.
- [7] A. Delcher, D. Harmon, S. Kasif, O. White, S. Salzberg, Improved microbial gene identification with GLIMMER *Nucleic Acids Research*, Vol. 27, No 23, 1999.
- [8] R. Gwadera, M. Atallah, and W. Szpankowski, Reliable detection of episodes in event sequences, In *Third IEEE International Conference on Data Mining*, pages 67-74, Melbourne Florida.
- [9] J. Han, J. Pei, Y. Yin, R. Mao, Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach, *Data Mining and Knowledge Discovery*, 8, 53-87, 2004
- [10] S. Katz, Estimation of probabilities from sparse data for the language model component of a speech recognizer, *IEEE Transactions on Acoustics, Speech and Signal Processing*, 35(3):400-401, March 1987.
- [11] H. Mannila, H. Toivonen, and A. Verkamo, Discovery of frequent episodes in event sequences, *Data Mining and Knowledge Discovery*, 1(3), 241-258, 1997.
- [12] A. Martin, G. Seroussi and M. Weinberger, *IEEE Transaction on Information Theory*, Vol. 50, No. 7, July 2004.
- [13] F. Jelinek, R. Mercer, Interpolated estimation of Markov source parameters from sparse data, *Proceedings of Workshop on Pattern Recognition in Practice*, pages 381-397, 1980.
- [14] M. Régnier and W. Szpankowski (1998), On pattern frequency occurrences in a Markovian sequence, *Algorithmica*, 22, 631-649.
- [15] J. Rissanen, A Universal Data Compression System, *IEEE Trans. Inform. Theory*, Vol. IT-29, No. 5, pp 656-664, 1983
- [16] J. Rissanen, Fast Universal Coding with Context Models, *IEEE Transactions on Information Theory*, Volume 45, No. 4, 1065-1071, May 1999
- [17] E. Ristad and R. Thomas, Nonuniform Markov Models, *International Conference on Acoustics, Speech, and Signal Processing*, Munich, Germany, April 20-24, 1997.
- [18] S. Salzberg, A. Delcher, S. Kasif, O. White, Microbial gene identification using interpolated Markov models, *Nucleic Acids Research*, Vol. 26, No 2, 1998.
- [19] W. Szpankowski (2001), *Average Case Analysis of Algorithms on Sequence*, John Wiley, New York.
- [20] M. Weinberger, J. Rissanen, and M. Feder, A universal finite memory source, *IEEE Trans. Inform. Theory*, vol. IT-41, pp. 643-652, May 1995. 48

- [21] F. Willems, Y. Shtarkov, and T. Tjalkens, The context-tree weighting method: Basic properties, *IEEE Trans. Inform. Theory*, vol. IT-41, pp. 653–664, May 1995 John Wiley, New York.