SYMBOLIC TO NUMERICAL CONVERSION OF DNA SEQUENCES USING FINITE-CONTEXT MODELS

Armando J. Pinho, Diogo Pratas, Paulo J. S. G. Ferreira and Sara P. Garcia

Signal Processing Lab, IEETA / DETI
University of Aveiro, 3810–193 Aveiro, Portugal
{ap,pratas,pjf,spgarcia}@ua.pt

ABSTRACT

Symbolic sequences can be analysed using two main approaches. One is by means of algorithms specifically designed for processing symbolic sequences. The other uses signal processing techniques, after converting the sequence from symbols to numbers. The latter approach depends on the availability of meaningful numerical representations of the sequences. In this paper, we present a technique that uses finite-context models to generate numerical information sequences from the DNA symbolic data. We give some examples that illustrate the method and show that these information sequences may reveal important structural properties of the DNA sequences. Moreover, the proposed approach is fast, allowing a quick bird's-eye view of whole chromosomes, with the aim of locating potentially interesting regions.

1. INTRODUCTION

DNA data sequences are symbolic and as such many of the tools that have been applied in their analysis are symbol-based (see, for example, [1, 2]). However, there is another large and rich set of tools available for the processing and analysis of numerical sequences. In principle, these tools can also be applied to DNA data. In fact, Fourier methods, correlation techniques and multi-resolution wavelet analysis have been used for studying DNA sequences for almost twenty years. Their application depends on several symbolic to numerical mappings, such as those involving indicator sequences [3], DNA walks [4], the vertices of a regular tetrahedron [5], or complex representations [6], just to name a few (see [7] for a review). Among the problems that are usually addressed using signal processing techniques, we point out those related to the discovery of short and long range correlations (see, for example, [3, 4, 8]) and the unveiling of periodicities (some examples can be found in [5, 9]).

In this paper, we address the problem of converting DNA data sequences into numerical sequences, through the generation of what are called "information sequences", "complexity sequences" or "complexity profiles". The theory behind these complexity profiles goes back to the works of several researchers in the 60's, such as Solomonoff, Kolmogorov, Chaitin and Wallace *et al.*, and is tightly related to the area of data compression [10]. The profiles are obtained using compression algorithms, because the size of the bitstream generated by a compression algorithm can be viewed as an upper bound of the Kolmogorov complexity of the compressed object.

In this context, the work of Allison *et al.* [11] has been of particular interest, because they have been trying to relate the information content of a DNA sequence (obtained by means of the per symbol code length generated by the encoder) with important characteristics of the DNA sequences. These information sequences were first presented in [11] and, more recently, in [12], in which they are suggested as a tool for the comparative analysis of long DNA sequences.

Most of the algorithms that have been proposed for compressing DNA sequences rely on particular aspects that characterize DNA, such as the existence of long exact or approximate repetitions, inverted complemented repeats and periodicities. DNA is highly non-stationary, with zones of low and high information content alternating frequently. This alternation of complexity is modeled by most DNA compression algorithms by a low order Markov chain model for the high entropy regions and by a Lempel-Ziv dictionary based approach for the repetitive, low entropy, regions.

In fact, finite-context (Markov) modelling has been used by most DNA sequence compression methods as a secondary, fall back solution. Usually, a low-order model (order-2 or order-3) is called into action when the main technique, based on dictionaries, fails to provide competitive results. However, the ability of finite-context modelling to represent DNA data sequences seems to go beyond a simple secondary role, as shown in recent work [13, 14], where the combination of two finite-context models led to encouraging results.

In this paper, we further explore the idea of using multiple finite-context models of several orders, working competitively, in this case for obtaining numerical sequences that capture important characteristics of the original DNA sequence, and that can be processed afterward using well established signal processing techniques. These numerical sequences represent the per DNA base information content (in Shannon's sense) and also the variation of model depth along the sequence. Among other possible applications of the approach presented in this paper, we point out the possibility of easily browsing along a chromosome, searching for regions potentially worthing further analysis.

2. FINITE-CONTEXT MODELLING OF DNA SEQUENCES

Consider an information source that generates symbols, s, from the alphabet $\mathscr{A} = \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}$. Also, consider that the information source has already generated the sequence of n symbols $x_{1..n} = x_1 x_2 \ldots x_n$, $x_i \in \mathscr{A}$. A finite-context model assigns probability estimates to the symbols of the alphabet, regarding the next outcome of the information source, according to a conditioning context computed over a finite and fixed number, $k > 0$, of the most recent past out-

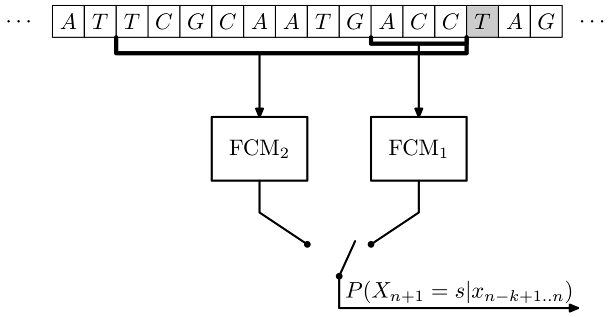
··· A T T C G C A A T G A C C T A G ···
FCM₂
FCM₁
$P(X_{n+1} = s|x_{n-k+1..n})$

Figure 1: Example of setup using two competing finite-context models. The probability of the next outcome, X_{n+1}, is conditioned by the $k = k_1$ or $k = k_2$ last outcomes, depending on the finite-context model chosen for encoding that particular DNA block. In this example, $k_1 = 3$ and $k_2 = 11$.

comes $x_{n-k+1..n} = x_{n-k+1} \dots x_n$ (order-k finite-context model) [15–17]. The number of conditioning states of the model is $|\mathscr{A}|^k$, where $|\mathscr{A}|$ denotes the size of the alphabet.

The probability estimates, $P(X_{n+1} = s|x_{n-k+1..n}), \forall_{s \in \mathscr{A}}$, are usually calculated using symbol counts that are accumulated while the sequence is processed. Thus, they depend not only on the past k symbols but also on n, i.e., these probability estimates are generally time-varying.

The theoretical per symbol information content average provided by the finite-context model after having processed n symbols is given by

$$H_{k,n} = -\frac{1}{n} \sum_{i=0}^{n-1} \log_2 P(X_{i+1} = x_{i+1}|x_{i-k+1..i}) \quad \text{bpb}, \tag{1}$$

where “bpb” stands for “bits per base”. Recall that the entropy of any sequence of four symbols is limited to two bits per symbol, a value that is obtained when the symbols are independent and equally likely.

In practice, the probability that the next outcome, X_{n+1}, is $s \in \mathscr{A}$, is obtained using the estimator

$$P(X_{n+1} = s|x_{n-k+1..n}) = \frac{C(s|x_{n-k+1..n}) + \alpha}{C(x_{n-k+1..n}) + 4\alpha}, \tag{2}$$

where $C(s|x_{n-k+1..n})$ represents the number of times that, in the past, the information source generated symbol s having $x_{n-k+1..n}$ as the conditioning context and where

$$C(x_{n-k+1..n}) = \sum_{a \in \mathscr{A}} C(a|x_{n-k+1..n}) \tag{3}$$

is the total number of events that occurred so far in association with context $x_{n-k+1..n}$. Parameter α controls how much probability is assigned to unseen (but possible) events, and plays a key role in the case of high order models. When k is large, the number of conditioning states, 4^k, is high, which implies that statistics have to be estimated using only a few observations.

For the multiple competing case, the several finite-context models are continuously updated, but only the best one is used for encoding a given region. For convenience, the DNA sequence is partitioned into non-overlapping blocks of fixed size, which are then encoded by the best model. Figure 1 gives an example of a setup using two competing models, one of depth $k_1 = 3$ and the other with depth $k_2 = 11$.

3. EXPERIMENTAL RESULTS

In this section, we provide some results obtained with the multiple competing finite-context models described previously. Two DNA sequences have been used for illustrating the method, namely the chromosomes 1 and 3 of the *Saccharomyces cerevisiae* (yeast) organism. Each of the sequences was processed using eight competing finite-context models with context depths $k = 2,4,6,8,10,12,14,16$. The decision of which depth to use was taken on a block by block basis, using blocks of ten DNA bases.

The probabilities associated to the finite-context models were estimated using (2), with $\alpha = 1$ (corresponding to Laplace's estimator) for model orders $k = 2,4,6,8,10$ and with $\alpha = 0.05$ for model orders $k = 12,14,16$. As previously mentioned, the value of α is not of much importance for low-order models, but it is fundamental in high-order cases. Note that, when the order of the model is high, the number of times that a given context occurs is generally small, rendering the estimation of the probability strongly dependent of α. According to our experience, $\alpha = 0.05$ provides, on average, good results.

When the model goes along the sequence, it is able to gather statistics and to “learn” its characteristics. Therefore, if after seeing a particular pattern the same (or approximate) pattern is seen again, then the information content will be lower than when it was observed for the first time. In other words, if some pattern is repeated, the information content associated with the consecutive appearances of the pattern will decrease (for each additional appearance the surprise will be lower, which, according to Shannon's information, leads to less bits of information). However, this procedure is directional, i.e., if the sequence is processed in one direction, then some characteristics may not be unveiled, the same happening if the opposite direction is used. To eliminate this directional dependency we ran the model first in one direction, to obtain one information sequence. Then, we ran the model in the opposite direction to obtain another. The final information sequence was obtained by picking the minimum value of the two at each point.

Figures 2 and 3 display several curves obtained using the DNA data sequences of chromosomes 1 and 3 of the *S. cerevisiae*. The top row shows the information content along these two chromosomes when the sequence is processed from left to right (5' to 3', using the notation of molecular biology). The row below displays the corresponding curves after the sequences have been processed from right to left (3' to 5'). As can be clearly seen, different structures are revealed depending on the direction of processing. When the two directional curves are combined (by picking the minimum values of each one), we obtain the curves shown in the third row of Figs. 2 and 3.

The last row of the graphics of Figs. 2 and 3 shows how the order of the finite-context model varies along the sequences (these have been obtained by averaging the two directional curves). As can be observed, although these curves seem to be correlated with the information sequences, some features appear only in one of them, suggesting that they are both potentially useful.

All of the curves displayed in Figs. 2 and 3 have been low-pass filtered. Filtering was done by averaging the samples using a Blackman window of size 401 centered at the sample under consideration. Varying the filter window size

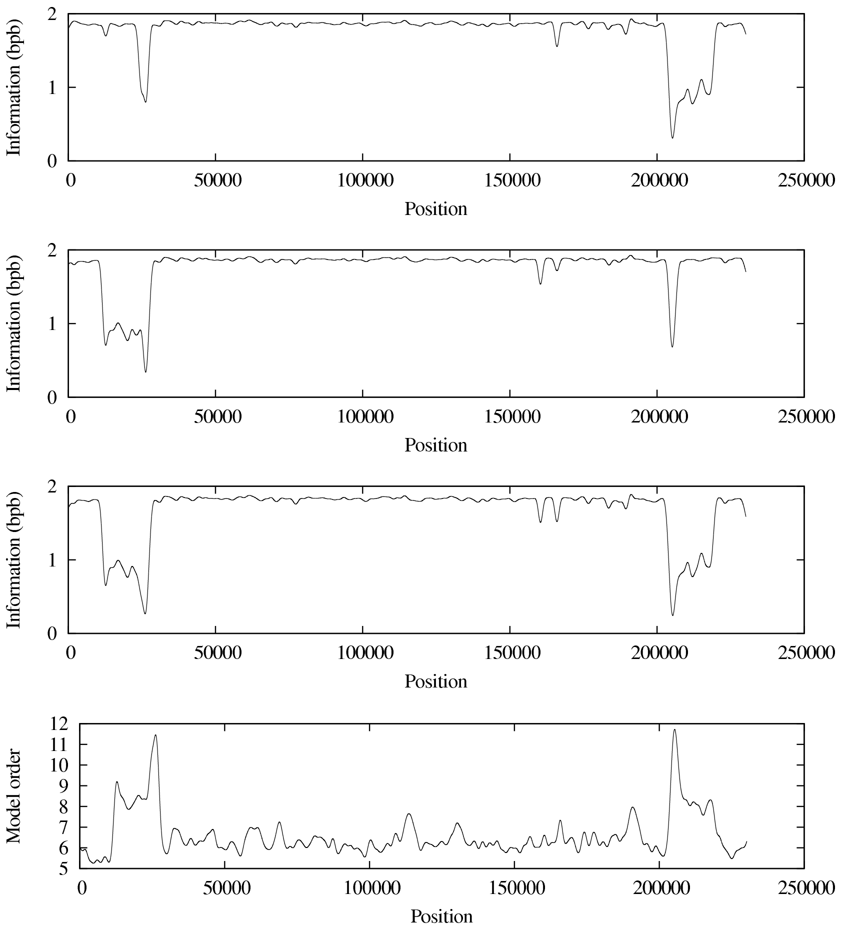

Figure 2: Plots of the information content and model order for chromosomes 1 of *S. cerevisiae*. The first row shows the information content when the sequence is processed from the left to the right. The second row shows the result when processing is done in the opposite direction. A combined version of both, using the minimum value of each one, is shown in the third row. Finally, the bottom row shows the order of the model along the sequence (based on the average value obtained by processing in both directions).

allows to conceal (larger windows) or unveil (smaller windows) different structures. Figure 4 shows the combined information profile of chromosome 1 filtered using window sizes of 21, 101, 201 and 1001 samples.

4. DISCUSSION AND CONCLUSION

There is a large number of powerful signal processing techniques that might be of great value for the field of molecular biology. However, for these techniques to be useful, the symbolic sequences need first to be converted into meaningful numerical sequences. There are several methods through which this can be accomplished, from indicator sequences to mappings into the complex plane.

In this paper a different viewpoint was taken, which brought into play the information content of the sequence. Although the idea of generating these information sequences is not new, to the best of our knowledge it is the first time that they are generated using multiple competing finite-context models.

As shown in the presented examples, these information profiles are of interest because they reveal structures inside the chromosomes, structures that are often associated with regulatory functions of DNA. For example, the patterns that appear near the beginning and end of the combined information curve of chromosome 1, reveling an almost identical profile, although reflected, are telomeric repeats called W', flanked by DNA sequences closely related to the yeast *FLO1* gene [18]. Other interesting structures can also be easily identified, depending on the scale used. In this work, we used a simple low-pass filter for being able to observe the profiles at different scales. However, much more sophisticated tools could be used for this purpose, such as wavelets.

The method that we propose has the great advantage of being fast. It is possible to process one of the largest human chromosomes (more than 200 million bases) in about one hour using a simple laptop. Therefore, for a certain DNA sequence, with this method it is possible to have a quick indication of regions of the sequence that might be worth looking at in more detail.

5. ACKNOWLEDGMENT

This work was supported in part by the grant with the COMPETE reference FCOMP-01-0124-FEDER-010099 (FCT, Fundação para a Ciência e Tecnologia, reference PTDC/EIA-

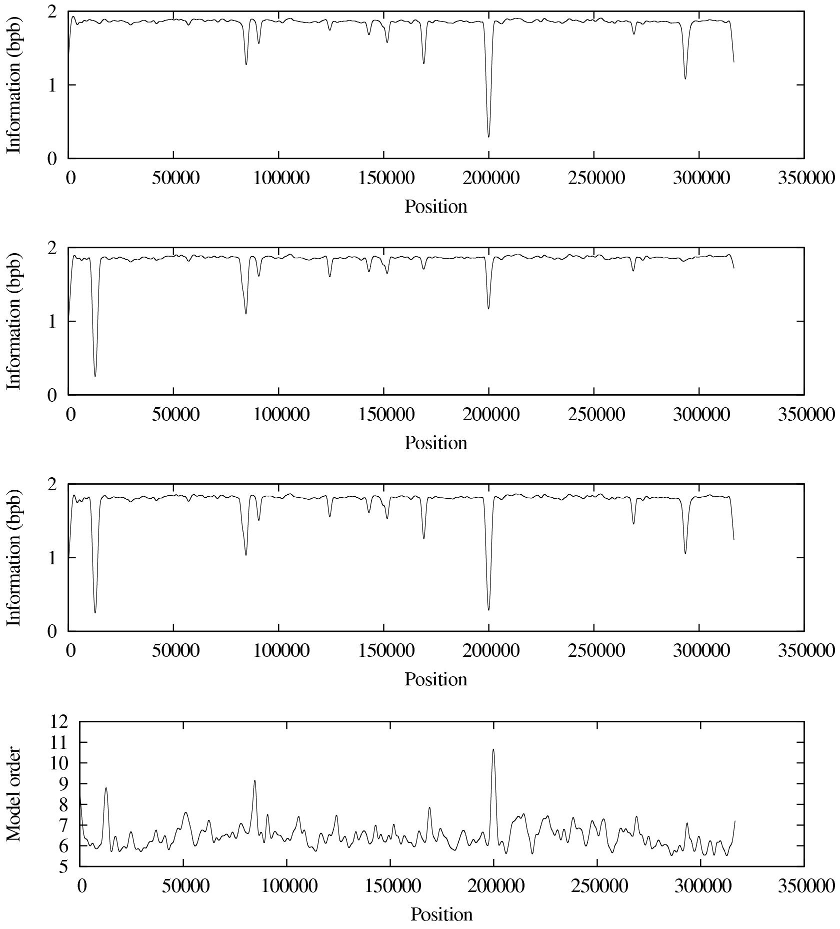

Figure 3: Similar plots as those of Fig. 2, but in this case for chromosome 3 of *S. cerevisiae*.

EIA/103099/2008). Sara P. Garcia acknowledges funding from the European Social Fund and the Portuguese Ministry of Science, Technology and Higher Education.

REFERENCES

[1] D. Gusfield, *Algorithms on strings, trees, and sequences: computer science and computational biology*, Cambridge University Press, Cambridge, 1997.

[2] R. C. Deonier, S. Tavaré, and M. S. Waterman, *Computational genome analysis — an introduction*, Springer, 2005.

[3] R. F. Voss, "Evolution of long-range fractal correlations and $1/f$ noise in DNA base sequences," *Physical Review Letters*, vol. 68, no. 25, pp. 3805–3808, June 1992.

[4] C.-K. Peng, S. V. Buldyrev, A. L. Goldberger, S. Havlin, F. Sciortino, M. Simons, and H. E. Stanley, "Long-range correlations in nucleotide sequences," *Nature*, vol. 356, pp. 168–170, Mar. 1992.

[5] D. Anastassiou, "Frequency-domain analysis of biomolecular sequences," *Bioinformatics*, vol. 16, no. 12, pp. 1073–1081, 2000.

[6] P. D. Cristea, "Large scale features in DNA genomic signals," *Signal Processing*, vol. 83, pp. 871–888, 2003.

[7] V. Afreixo, P. J. S. G. Ferreira, and D. Santos, "Fourier analysis of symbolic data: a brief review," *Digital Signal Processing*, vol. 14, no. 6, pp. 523–530, Nov. 2004.

[8] S. V. Buldyrev, A. L. Goldberger, S. Havlin, R. N. Mantegna, M. E. Matsa, C.-K. Peng, M. Simons, and H. E. Stanley, "Long-range correlation properties of coding and noncoding DNA sequences: GenBank analysis," *Physical Review E*, vol. 51, no. 5, pp. 5084–5091, May 1995.

[9] D. Anastassiou, "Genomic signal processing," *IEEE Signal Processing Magazine*, vol. 18, no. 4, pp. 8–20, 2001.

[10] M. Li, X. Chen, X. Li, B. Ma, and P. M. B. Vitányi, "The similarity metric," *IEEE Trans. on Information Theory*, vol. 50, no. 12, pp. 3250–3264, Dec. 2004.

[11] L. Allison, T. Edgoose, and T. I. Dix, "Compression of strings with approximate repeats," in *Proc. of Intelligent Systems in Molecular Biology, ISMB-98*, Montreal, Canada, 1998, pp. 8–16.

[12] T. I. Dix, D. R. Powell, L. Allison, J. Bernal, S. Jaeger, and L. Stern, "Comparative analysis of long DNA sequences by per element information content using different contexts," *BMC Bioinformatics*, vol. 8, no. Suppl. 2, pp. S10, 2007.

[13] A. J. Pinho, A. J. R. Neves, and P. J. S. G. Fer-

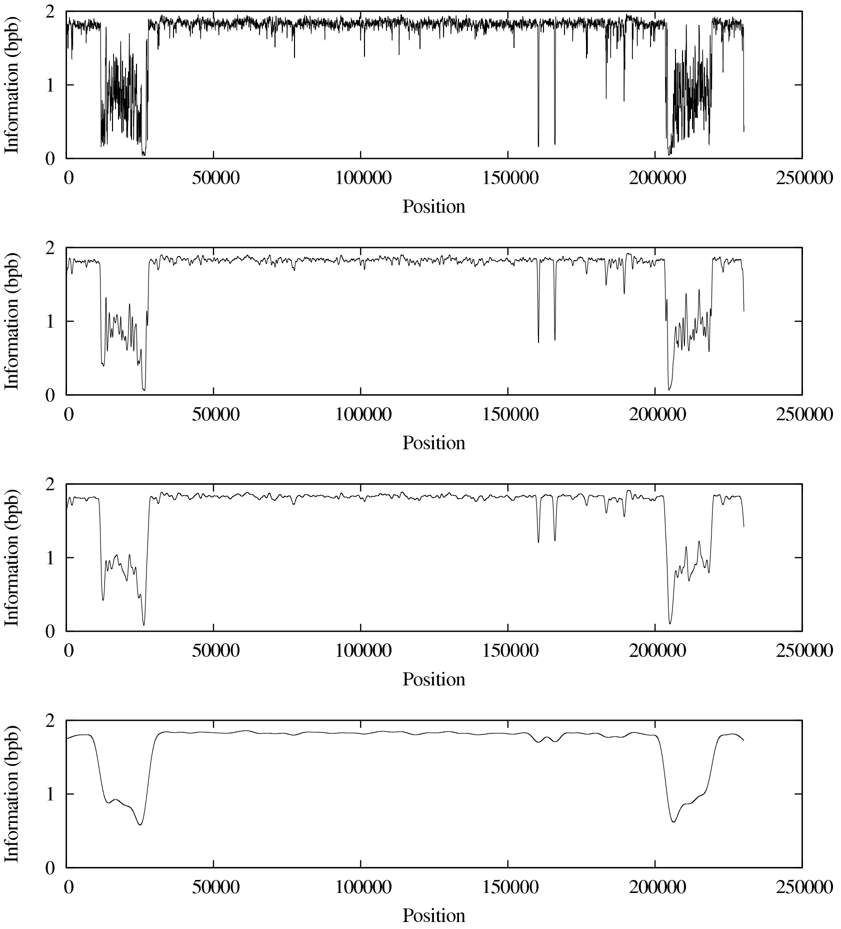

Figure 4: Plots of the combined information content of chromosome 1 of the *S. cerevisiae* organism using different filter window sizes. From top to bottom, window size of 21, 101, 201 and 1001 samples. Varying this parameter allows looking for structural information at several scales.

reira, "Inverted-repeats-aware finite-context models for DNA coding," in *Proc. of the 16th European Signal Processing Conf., EUSIPCO-2008*, Lausanne, Switzerland, Aug. 2008.

[14] A. J. Pinho, A. J. R. Neves, C. A. C. Bastos, and P. J. S. G. Ferreira, "DNA coding using finite-context models and arithmetic coding," in *Proc. of the IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, ICASSP-2009*, Taipei, Taiwan, Apr. 2009.

[15] T. C. Bell, J. G. Cleary, and I. H. Witten, *Text compression*, Prentice Hall, 1990.

[16] D. Salomon, *Data compression - The complete reference*, Springer, 4th edition, 2007.

[17] K. Sayood, *Introduction to data compression*, Morgan Kaufmann, 3rd edition, 2006.

[18] H. Bussey, D. B. Kaback, W.-W. Zhong, D. T. Vo, M. W. Clark, N. Fortin, J. Hall, et al., "The nucleotide sequence of chromosome I from *Saccharomyces cerevisiae*," *Proceedings of the National Academy of Sciences USA*, vol. 92, pp. 3809–3813, Apr. 1995.