Employing compact intra-genomic language models to predict genomic sequences and characterize their entropy

Sérgio Deusdado[1] **and Paulo Carvalho**[2]

[1]CIMO - Mountain Research Centre, Polytechnic Institute of Bragança, Portugal

[2]Department of Informatics, School of Engineering, University of Minho, Braga, Portugal

Abstract Probabilistic models of languages are fundamental to understand and learn the profile of the subjacent code in order to estimate its entropy, enabling the verification and prediction of "natural" emanations of the language. Language models are devoted to capture salient statistical characteristics of the distribution of sequences of words, which transposed to the genomic language, allow modeling a predictive system of the peculiarities and regularities of genomic code in different inter and intra-genomic conditions. In this paper, we propose the application of compact intra-genomic language models to predict the composition of genomic sequences, aiming to achieve valuable resources for data compression and to contribute to enlarge the similarity analysis perspectives in genomic sequences. The obtained results encourage further investigation and validate the use of language models in biological sequence analysis.

Keywords language models, DNA entropy estimation, genomic sequences modeling

1 Introduction

Language models aim to capture the context of a language based on the study and computation of the probabilities of its patterns [1], developing models to infer the rules behind the successions of its segments, i.e. words, n-grams, sounds, codons, etc. Hidden Markov Models (HMMs) also rely on probabilistic models and are widely used in bioinformatics for gene prediction and profiling of sequences [2].

Entropy measures of DNA sequences estimate their randomness or, inversely, their repeatability [3]. In the field of genomic data compression, fundamentally based on the comprehension of the regularities of genomic language and entropy

[1] Corresponding author: *sergiod@ipb.pt*

estimation, language models appear as a promising methodology to characterize the genomic linguistics and to provide predictive models for data compression [4][5] [6], as well as revealing new approaches for sequence similarity analysis [7]. Statistical language models are widely used in speech recognition [8] and have been successfully applied to solve many different information retrieval problems [9]. A good review on statistical language modeling is presented by Rosenfeld in [10].

Currently, the Biological Language Modeling Toolkit is a good example of the interest of this field of investigation, developed by the Center for Biological Language Modeling. This toolkit consists on a compilation of various algorithms that have been adapted to biological sequences from language modeling, and specifically it is oriented to uncover the "protein sequence language". The toolkit is publicly available at the following URL: http://flan.blm.cs.cmu.edu/12/HomePage.

2 Language Models

Language modeling is the art of determining the probability of a word sequence $w_1...w_n$, $P(w_1...w_n)$ [10]. This probability is typically divided into its component probabilities:

$$P(w_1...w_i) = P(w_1) \times P(w_2|w_1) \times ... \times P(w_i|w_1...w_{i-1}) \qquad (2.1)$$

$$= \prod_{i=1}^{n} P(w_i \mid w_1, w_2, ..., w_{i-1})$$

Since it may be difficult to compute the probability $P(w_i|w_1...w_{i-1})$ for large i, it is typically assumed that the probability of a word depends on only the two previous words. Thus, that trigram assumption can be written as:

$$P(w_i|w_1...w_{i-1}) \approx P(w_i|w_{i-2}w_{i-1}) \qquad (2.2)$$

The trigram probabilities can then be estimated from their counts in a training corpus. Let $C(w_{i-2}w_{i-1}w_i)$ represent the number of occurrences of $w_{i-2}w_{i-1}w_i$ in our training corpus, and similarly for $C(w_{i-2}w_{i-1})$. Then, we can approximate:

$$P(w_i|w_{i-2}w_{i-1}) \approx C(w_{i-2}w_{i-1}w_i)\ C(w_{i-2}w_{i-1}) \qquad (2.3)$$

The most obvious extension to trigram models is to move to higher order n-grams, such as four-grams and five-grams. In genomic language modeling is usual to consider codons as words. Codons are three-letter words from the quaternary genomic alphabet {A, C, G, T}, resulting only 64 possible combinations. Thus, genomic language models generally use higher order n-grams to improve their efficiency.

Smoothing techniques are used to avoid zero probability n-grams which may occur from inadequate training data [11]. In fact, rare trigrams should also integrate the predictive model; therefore its probability, even low, must be greater than zero. On the other hand, smoothing affects high probabilities to be adjusted downward. Not only do smoothing methods generally prevent zero probabilities, but they also attempt to improve the accuracy of the model as a whole.

The most commonly used method for measuring language model performance is *perplexity*. In general, the *perplexity* of a n-gram language model is equal to the geometric average of the inverse probability of the words measured on test data:

$$\sqrt[n]{\prod_{i=1}^{n} \frac{1}{P(w_i \mid w_1 ... w_{i-1})}} \qquad (2.4)$$

Low *perplexity* of the model means high fidelity predictions. A language model assigning equal probability to 100 words would have perplexity 100. An alternative, but equivalent measure to *perplexity* is entropy, which is simply $\log_2$ of *perplexity*.

3 Developed Work

The developed work corresponds to a framework for entropy estimation and analysis of DNA sequences based on cooperative intra-genomic compact language models. These models will obtain a probability distribution of the next symbol at a given position, based on the symbols previously seen. Based on the experiments of Cao *et al.* [12], we choose to divide our approach into global and local models, combining their contribution to improve the efficiency of the multiparty predictive model. While global models consider the full extension of the analyzed sequences, local models only capture the probabilistic properties of a limited number of bases preceding the base(s) to predict, considering if necessary a variable displacement.

Our aim was to take advantage of the successive probability present mainly in repetitive regions of DNA sequences, as well as in non-repetitive regions where a stochastic model can be efficient too.

We used a backoff n-gram language model [13][14] implemented with arrays of values representing the most probable chain of codons to occur after each one of the 64 possible codons. Our models were not trained based on a corpus because the intention was to apply, subsequently, the resulting framework to an on-line genomic data compression algorithm. In this sense, the resulting compressed file must be self-contained, as the recalculation of probabilities in the decompression process relies only on the data included in the compressed file. Thus, the need for compact models, especially the global model because it is integrated in the compressed file. The local models are adaptive and evolve periodically as they receive new input from the history of the sequence already viewed, i.e. the significant portion of the sequence preceding the point of prediction. In this way, we produce intra-genomic and very compact models, expecting not to compromise the processing time of the predictive algorithm and, additionally, looking forward to include the essential part of the models in the resulting compressed file to help the initial predictions, when history is not available.

Experimental results, using a typical set of DNA sequences (see Table 4.1) used in DNA compressors as test corpus, showed that ten-grams/codons corresponds to the most appropriated order for global models, considering the tradeoff between model performance and computational cost. For local models, we used order twenty, not codon based but nucleotide based. Each model presents its prediction supported by an associated probability, reflecting the model's confidence. At the end, all local and global models are pooled to elect by a voting system the definitive prediction to be emitted. Independently of the model that produces the prediction, any prediction is just a symbol. Hence, if the prediction is made by a codon-based model only the first nucleotide of the predicted codon is considered.

Local Models

Local models use single letters (nucleotides) instead of codons and are of order thirty. Being adaptive, they are modified with the latter knowledge obtained from the already predicted - and eventually corrected - sequence. The local models used in our work are based on 1000 nucleotides context, not immediately before the nucleotide to predict but forming a window slid back in the sequence. We used two versions based on different displacements, one with 500 bp displacement and the other going backward 3000 bp. We used different types of local models to enlarge the possibilities of our prediction system, trying to take advantage on the best available knowledge, such as being aware of that most repetitions occur hundreds or thousands of bp after its last occurrence. Considering that some DNA regularities occur in the reverse complementary form, the so-called palindromes, we used complimentary local models based on reverse complementary history.

Global Models

Global models use codons and gather the probabilistic study of ten-grams. A global model based on the reverse complementary sequence was also produced. Global models are meant to be compact, as they will integrate a future compressed file of the DNA sequence. Our global models are based on tables, containing the most probable succession of codons to occur after each one of the 64 possible codons. Without increasing data complexity, it is possible to calculate global models for the three frames of a sequence. In this way, frame 1, 2 and 3 variants were also considered. These models can be consulted also for subsequences of codons, not necessarily initiated at order 0, using a backing off strategy. An example of a global model, upon analysis of the frame 1 of a sequence, is shown in Table 3.1.

Frame 1		Order										
		0	1	2	3	4	5	6	7	8	9	10
Code	Codon	Sucession of codons with highest probability										
1	AAA	1→	4	4	16	63	29	32	1	1	2	34
2	AAC	2→	12	12	57	23	44	43	5	4	1	31
…	…	…	…	…	…	…	…	…	…	…	…	…
63	GGT	63→	52	2	64	4	11	13	24	6	12	59
64	GGG	64→	11	13	24	6	12	59	17	55	32	4

Table 3.1. Example of a compact global model considering ten-grams.

Prediction Process

The test prototype considers six different models, described as follows:

M1 – regular global model;
M2 – global model based on the reverse complementary sequence;
M3 – regular local model considering 1000 previous symbols context and a retro-displacement of 3000 bases;
M4 – reverse complementary local model considering 1000 previous symbols context and a retro-displacement of 3000 bases;
M5 – regular local model considering 1000 previous symbols context and a retro-displacement of 500 bases;
M6 – reverse complementary local model considering 1000 previous symbols context and a retro-displacement of 500 bases.

A model emits a prediction based on order n when it contains the knowledge of a probable n-gram equal to the one at the end of the analyzed portion of the sequence. When there is a conflict between predictions of equal order, global models have priority and their predictions prevail as they derive from the complete sequence. If a global model produces a prediction of order ≥ 3 then the local models predictions are ignored. Each model votes for its predicted symbol, and in the end a probabilistic distribution emerges from a voting system where global models have more weight on final results than local models. Votes from local models are equal to the order used in the prediction, whereas the global models' orders used in the predictions are trebled. Table 3.2 shows an example of the election of a final prediction and its probability distribution based on the following individual predictions cases:

M1 – predicted A with order 5(x3);
M2 – predicted C with order 1(x3);
M3 – predicted T with order 3;
M4 – predicted A with order 1;
M5 – predicted C with order 2;
M6 – predicted T with order 1;

To prevent zero probability, non-voted symbols receive one vote.

Prediction	Votes by Model						Total	Probability distribution
	M1	M2	M3	M4	M5	M6		
A	15			1			16	62%
C		3			2		5	19%
T			3			1	4	15%
G							1	4%

Table 3.2. Demonstration of the voting system used to achieve the probability distribution.

4 Experimental Results

A test prototype was implemented to combine the predictions from the models described in the previous section in order to assess the predictive capability of our framework. The code was written in C language and compiled using *gcc* version 3.4.2, configured for maximal code optimization. Tests ran on a system based on Intel Pentium IV – 3,4GHz, 8KB L1 + 512 KB L2 cache, with 1GB RAM-DDR and a 250 GB HD. We tested our prototype on a dataset of DNA sequences typically used in DNA compression studies. The dataset includes 11 sequences: two chloroplast genomes (CHMPXX and CHNTXX); five human genes (HUMDYSTROP, HUMGHCSA, HUMHBB, HUMHDABCD and HUMHPRTB); two mitochondria genomes (MPOMTCG and MTPACG); and genomes of two viruses (HEHCMVCG and VACCG).

Table 4.1 contains the results obtained, considering the percentage of predictions that matched the corresponding symbol in the original sequence.

Sequence	***Length(bp)***	***% of correct predictions***
CHMPXX	121.024	29
CHNTXX	155.844	30
HEHCMVCG	229.354	27
HUMDYSTROP	38.770	26
HUMGHCSA	66.495	37
HUMHBB	73.323	28
HUMHDAB	58.864	29
HUMHPRTB	56.737	28
MPOMTCG	186.608	27
MTPACG	100.314	27
VACCG	191.737	28
Average	**116.279**	**29**

Table 4.1. Experimental results.

Considering the quaternary alphabet and carrying out a random prediction, it will be expectable, in theory, to obtain 25% of correct predictions, on average. Comparatively, the obtained results exhibit 29% of prediction correctness on average. However, the obtained results are satisfactory considering only intra-genomic study and the reduced size of the used models; moreover the high level of entropy inherent to DNA sequences justifies the quality of the results. HUMGHCSA is the sequence where our predictive model performed better because it is, within the tested set of sequences, the one with lower entropy, i.e. high redundancy, as we may confirm in the literature [4][15].

5 Conclusions and Future Work

Experimental results demonstrate a linear correlation facing the entropy of each tested sequenced based on the results of existing DNA data compressors [4][3][12][15]. Sequences with higher levels of entropy are more difficult to model and hence our models get modest results on their analysis. Global models capture the most significant patterns of the sequence and perform generally better. Local models revealed low utility in entropy estimation but they are important to complement predictions. Different sequences may need proper adjustments of the extension and the displacement of the local models to optimize their prediction capability. We believe that it would be useful to determine the profile of the sequence in advance in order to adaptively adjust the local model's parameterization. This will be addressed in future developments.

Our major goal was to test the potential and efficiency of language models as a complementary compression method for biological data compression, for instance, to complement dictionary based techniques. Consequently, our focus was mainly in regions of the sequences where linear patterns are sparse or exist repeatedly but with reduced extension. Additional work is needed to optimize all the models, especially the local ones, but the obtained results encourage further investigation.

6 References

1. Durbin R, Eddy S, Krogh A, Mitchison G (1998) Biological sequence analysis: probabilistic models of proteins and nucleic acids. Cambridge University Press, Cambrige.
2. Koski T (2001) Hidden Markov Models for Bioinformatics. Kluwer Academic Publishers.
3. Lanctot JK, Li M, Yang E (2000) Estimating DNA sequence entropy. SODA '00: Proceedings of the eleventh annual ACM-SIAM symposium on Discrete algorithms. pp. 409-418 Society for Industrial and Applied Mathematics, San Francisco, California, United States.
4. Loewenstern D, Yianilos PN (1997) Significantly Lower Entropy Estimates for Natural DNA Sequences. Data Compression Conference (DCC '97). p. 151.
5. Osborne M (2000) Predicting DNA Sequences using a Backoff Language Model, http://www.cogsci.ed.ac.uk/~osborne/dna-backoff.ps.gz

6. Venugopal KR, Srinivasa KG, Patnaik LM (2009) Probabilistic Approach for DNA Compression. Soft Computing for Data Mining Applications. pp. 279-289 Springer Berlin / Heidelberg.
7. Buehler EC, Ungar LH (2001) Maximum Entropy Methods for Biological Sequence Modeling. Workshop on Data Mining in Bioinformatics (with SIGKDD01 Conference). pp. 60-64 , San Francisco, CA, USA.
8. Jelinek F (1997) Statistical methods for speech recognition. MIT Press, Cambridge Mass..
9. Zhai C (2008) Statistical Language Models for Information Retrieval. Synthesis Lectures on Human Language Technologies. 1, 1-141.
10. Rosenfeld R (2000) Two Decades of Statistical Language Modeling: Where Do We Go from Here? Proceedings of the IEEE. 88, 1270-1278.
11. Chen SF, Goodman J (1998) An Empirical Study of Smoothing Techniques for Language Modeling. Harvard University.
12. Cao MD, Dix TI, Allison L, Mears C (2007) A Simple Statistical Algorithm for Biological Sequence Compression. 2007 Data Compression Conference (DCC'07). pp. 43-52 , Snowbird, UT, USA.
13. Katz S (1987) Estimation of probabilities from sparse data for the language model component of a speech recognizer. IEEE Transactions on Acoustics, Speech, and Signal Processing. 35, 400-40.
14. Kneser R, Ney H (1995) Improved backing-off for m-gram language modeling. IEEE Int. Conf. Acoustics, Speech and Signal Processing. pp. 181-184 IEEE, Detroit, USA.
15. Korodi G, Tabus I (2005) An efficient normalized maximum likelihood algorithm for DNA sequence compression. ACM Transactions on Information Systems. 23, 3-34.