Available online at www.sciencedirect.com
SCIENCE DIRECT®
Physica A 342 (2004) 294–300
PHYSICA A
www.elsevier.com/locate/physa

Dictionary-based methods for information extraction

A. Baronchelli[a], E. Caglioti[b], V. Loreto[a,*], E. Pizzi[c]

[a] *Physics Department, "La Sapienza" University, P.le A. Moro 5, 00185 Rome, Italy*
[b] *Mathematics Department, "La Sapienza" University, P.le A. Moro 2, 00198 Rome, Italy*
[c] *Istituto Superiore di Sanità, Viale Regina Elena 299, 00161 Rome, Italy*

Received 31 October 2003; received in revised form 16 January 2004

Abstract

In this paper, we present a general method for information extraction that exploits the features of data compression techniques. We first define and focus our attention on the so-called *dictionary* of a sequence. Dictionaries are intrinsically interesting and a study of their features can be of great usefulness to investigate the properties of the sequences they have been extracted from e.g. DNA strings. We then describe a procedure of string comparison between dictionary-created sequences (or *artificial texts*) that gives very good results in several contexts. We finally present some results on self-consistent classification problems.

PACS: 89.70; 87.10.+e

Keywords: Information extraction; Data compression; Sequence analysis

1. Introduction

Strings or sequences of characters appear in almost all sciences. Examples are written texts, DNA sequences, bits for the storage and transmission of digital data etc. When analyzing such sequences the main point is extracting the information they bring. For a DNA sequence this could help in identifying regions involved in different functions (e.g. coding DNA, regulative regions, structurally important domains) (for a recent review of computational methods in this field see Ref. [1]). On the other hand for a

* Corresponding author.
E-mail address: loreto@roma1.infn.it (V. Loreto).

0378-4371/$ - see front matter
doi:10.1016/j.physa.2004.01.072

written text one is interested in questions like recognizing the language in which the text is written, its author or the subject treated.

When dealing with information related problems, the natural point of view is that offered by information theory [2,3]. In this context the word information acquires a precise meaning which can be quantified by using the concept of entropy. Among several equivalent definitions of entropy the best one, for our purposes, is that of Algorithmic Complexity proposed by Chaitin, Kolmogorov and Solomonoff [4]: the Algorithmic Complexity of a string of characters is the length, in bits, of the smallest program which produces as output the string and stop afterward.

Though it is impossible, even in principle, to find such a program, there are algorithms explicitly conceived to approach such theoretical limit. These are the file compressors or zippers. In this paper we shall investigate some properties of a specific zipper, LZ77 [5], used as a tool for information extraction.

2. The dictionary of a sequence

It is useful to recall how LZ77 works. Let $x = x_1,\ldots,x_N$ be the sequence to be compressed, where x_i represents a generic character of sequence's alphabet. The LZ77 algorithm finds duplicated strings in the input data. The second occurrence of a string is replaced by a pointer to the previous string given by two numbers: a distance, representing how far back into the window the sequence starts, and a length, representing the number of characters for which the sequence is identical. More specifically the algorithm proceeds sequentially along the sequence. Let us suppose that the first n characters have been codified. Then the zipper looks for the largest integer m such that the string $x_{n+1},\ldots,x_{n+m}$ already appeared in $x_1,\ldots,x_n$. Then it codifies the string found with a two-number code composed by: the distance between the two strings and the length m of the string found. If the zipper does not find any match then it codifies the first character to be zipped, x_{n+1}, with its name. This eventuality happens for instance when codifying the first characters of the sequence, but this event becomes very infrequent as the zipping procedure goes on.

This zipper has the following remarkable property: if it encodes a text of length L emitted by an ergodic source whose entropy per character is h, then the length of the zipped file divided by the length of the original file tends to h when the length of the text tends to infinity [6]. In other words LZ77 does not encode the file in the best way but it does it better and better as the length of the file increases. Usually, in commercial implementations of LZ77 (like for instance *gzip*), substitutions are made only if the two identical sequences are not separated by more than a certain number n of characters, and the zipper is said to have a n-long sliding window. The typical value of n is 32768. The main reason for this restriction is that the search in very large buffers could be not efficient from the computational time point of view. A restriction is often given on the length of a match, too, avoiding substitution of repeated subsequences shorter than 3 characters.

We define *dictionary* [7] of a string the whole set of sub-sequences that are substituted with a pointer by LZ77 and we refer to these sub-sequences as dictionary's

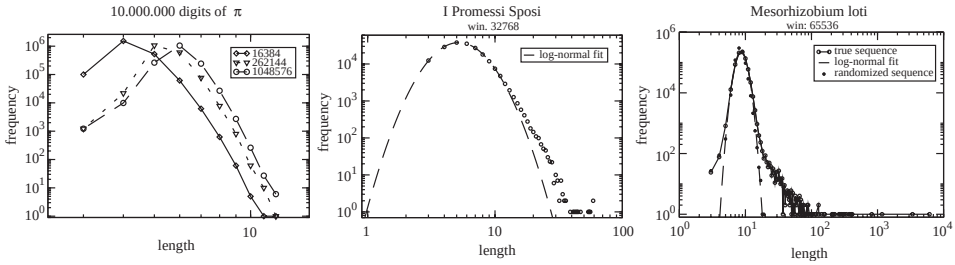

Fig. 1. Frequency-length distributions for words in the dictionaries of different sequences. Left: The sequence of the first 10^6 characters of π. Dictionaries extracted with different window lengths. Center: The Italian book "I Promessi Sposi", with a log-normal fit of the peak of the distribution. Right: *Mesorhizobium loti* original and reshuffled sequences, with the log-normal fit of the peak.

words. From the previous discussion it is clear that the same word can appear several times in our dictionary (the multiplicity being limited by the length of the sequence). Moreover, the structure of a dictionary is determined by the size of the LZ77 sliding window. In particular, it has been shown [6,8] that the average word length l found by an n-long sliding window LZ77 goes asymptotically as $l = \log n/h$, where h is the entropy of the (ergodic) source that emitted the sequence. It follows that the size of the sliding window does not affect the number of characters in the dictionary, but the way they are combined into words.

In Fig. 1 the frequency-length distributions for the words in the dictionaries of several sequences of increasing complexity are presented. In each figure the number of words of any length is plotted. For the sequence of digits of π (which can be assumed to be a sequence of realizations of independent and identically distributed random variables) the spectra obtained for three different sizes of the LZ77 sliding window are presented. As expected the peak of the distribution grows with the window's size. In the central plot the dictionary of the Italian book "I Promessi Sposi" is analyzed. In this case, while the peak is well fitted by a log-normal distribution (i.e., a Gaussian in logarithmic scale), several very long words appear. The presence of long words becomes crucial in the dictionary extracted by the DNA sequence of *Mesorhizobium loti* in the right plot. Here we compare the dictionary extracted from the true sequence with the one obtained from its randomization. As expected, long words are absent in the dictionary of the reshuffled sequence.

Since a genome is composed of regions coding for proteins (genes) and of intergenic non-coding tracts, we have analyzed in more detail the contribution of these parts to the distributions of repeated "words". In Fig. 2 results obtained in the case of *Escherichia coli* genome are reported. This genome is approximately 4.500.000 bp long; the 87% belongs to coding regions (see dotted line in the figure on the right). In the figure on the left, the frequency-length distributions for the entire genome and for the coding tracts are reported. The two distributions appear as completely overlapped up to 20 bp of length, while for the next lengths they deviate from each other. This fact is highlighted in the figure on the left, where the fraction of words of each length coming from

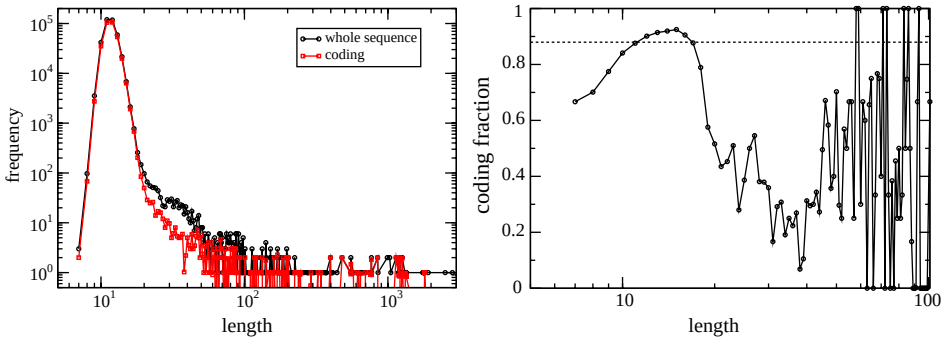

Fig. 2. Fraction of words extracted from coding regions (*Escherichia coli*). This dictionaries were extracted giving LZ77 the possibility of finding repeated sequences in the whole string. Words of lengths between 20 and 90 characters are found to belong mainly to non-coding regions.

coding regions is reported. It is clearly visible that within a range of approximately 20–90 bp, most words come from non-coding tracts. We observed an analogous behavior in the *Vibrio cholerae* second chromosome analysis (data not shown). It is a well known fact that non-coding sequences are characterized by the presence of repeated "words", however, at least for the analysed prokaryotic genomes, our results seem to suggest that these tracts are not more repetitive than genes but, more precisely, that they are characterized by repeated words longer than those occurring within coding parts. Furthermore these preliminary results suggest our approach as an useful tool to study genomes and their organization.

3. Dictionary-based self classification of corpora

Data compression schemes can be also used to compare different sequences. In fact it has been shown [9–11] that, compressing with LZ77 a file B appended to a file A, it is possible to define a remoteness between the two files. More precisely the difference between the length of the compressed file A+B and the length of the compressed file A, all divided by the length of the file B, can be related to the cross entropy [1] between the two files [12]. This method is strictly related to the algorithm by Ziv and Merhav [13] which allows to obtain a rigorous estimate of the cross entropy between two files A and B by compressing, with an algorithm very similar to LZ77, the file B in terms of the file A. In Ref. [11] experiments of language recognition, authorship attribution and language classification are performed exploiting the commercial zipper *gzip* to implement the technique just discussed. In this paper we use a natural extension of the method used in Ref. [11], devised to measure directly the cross entropy between A and B: in particular the LZ77 algorithm only scans the B part and looks for matches

[1] With the term cross-entropy between two strings we shall always refer in this paper to an estimate of the true cross-entropy between the two ergodic sources from which A and B have been generated.

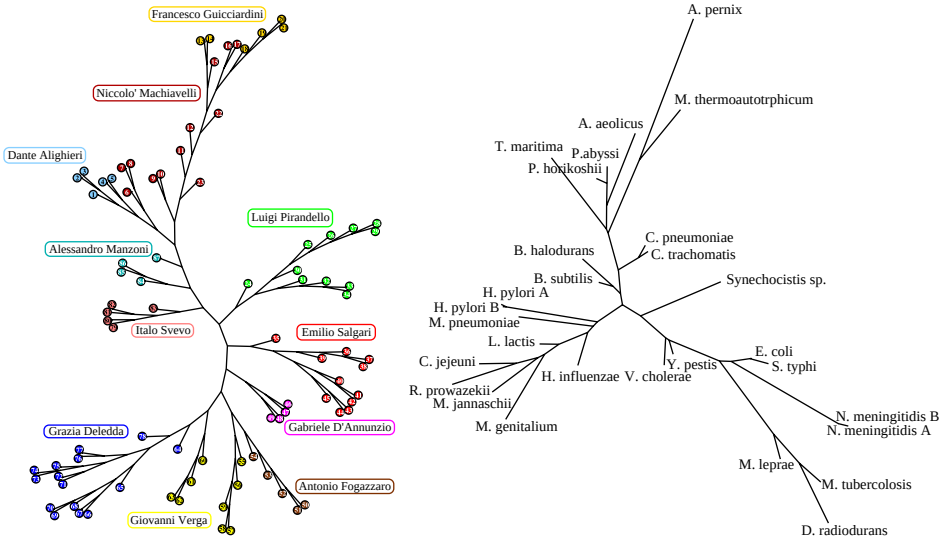

Fig. 3. Self-consistent classification. Left: a tree obtained from a corpus of 87 works of 11 Italian writers. Right: a species similarity tree for 27 procariotes. Both trees have been obtained from distance matrices constructed with the artificial texts comparison method.

only in the A part. In experiments of features recognition (for instance language or authorship) a text X is compared with each text A_i of a corpus of known texts. The closest A_i sets the feature of the X text (i.e., its language or author). In classification experiments, on the other hand, one has no a priori knowledge of any texts and the classification is achieved by the construction of a matrix of the distances between pairs of sequences. A suitable tree representation of this matrix can be obtained using techniques mutuated from phylogenetics. It must be underlined that, for self-consistent classification problems, a true mathematical distance is needed (see for previous important related works on this subject [4,14] and for a general discussion [15]).

Our idea (see also [7]) is that of creating *artificial texts* by appending words randomly extracted from a dictionary and to compare artificial texts instead of the original sequences. The comparison of artificial texts is made using the modified version of LZ77 discussed above. One of the biggest advantages of our artificial text method is the possibility of creating an ensemble of artificial texts all representing the same original sequence, thus enlarging the original set of sequences. Comparing artificial texts we performed the same experiments described in Ref. [11] obtaining better results.

In Fig. 3 we present a linguistic tree representing the self-classification of a corpus of 87 texts belonging to 11 Italian authors [16]. The texts belonging to the same author clusterize quite well, with the easily-explainable exception of the Machiavelli and Guicciardini clusters. The other tree presented in Fig. 3 is obtained by a whole-genome comparison of 27 prokaryotic genomes. This kind of analysis are now definitely possible thanks to the availability of completely sequenced genomes (See for a similar

approach [17]). Our results appear as comparable with those obtained through other completely different "whole-genome" analysis (see, for instance, [18]). Closely related species are correctly grouped (as in the case of *E. coli* and *S. typhimurium*, *C. pneumoniae* and *C. trachomatis*, *P. abyssi* and *P. horikoshii*, etc.), and some main groups of organisms are identified. It is known that the mono-nucleotide composition is a specie-specific property for a genome. This compositional property could affect our method: namely two genomes could appear as similar simply because of their similar C+G content. In order to rule out this hypothesis we performed a new analysis after shuffling genomic sequences and we noticed that the resulting new tree was completely different with respect to the one based on real sequences.

In conclusion we have defined the dictionary of a sequence and we have shown how it can be helpful for information extraction purposes. Dictionaries are intrinsically interesting and a statistical study of their properties can be a useful tool to investigate the strings they have been extracted from. In particular new results regarding the statistical study of DNA sequences have been presented here. On the other hand, we have proposed an integration of the string comparison procedure presented in Ref. [11] that exploits dictionaries by means of artificial texts. This method gives very good results in several contexts and we have focused here on self-classification problems, showing two similarity trees for corpora of written texts and DNA sequences.

References

[1] T. Jiang, Y. Xu, M. Zang (Eds.), Current Topics in Computational Molecular Biology, MIT Press, Cambridge, MA, London, England.
[2] C.E. Shannon, A mathematical theory of communication, The Bell System Tech. J. 27 (1948) 379–423, 623–656.
[3] W.H. Zurek (Ed.), Complexity, Entropy and Physics of Information, Addison-Wesley, Redwood City, 1990.
[4] M. Li, P.M.B. Vitanyi, An Introduction to Kolmogorov Complexity and its Applications, 2nd Edition, Springer, Berlin, 1997.
[5] A. Lempel, J. Ziv, A universal algorithm for sequential data compression, IEEE Trans. Inform. Theory IT-23 (1977) 337–343.
[6] A.D. Wyner, J. Ziv, The sliding-window Lempel-Ziv algorithm is asymptotically optimal, Proc. IEEE 82 (1994) 872–877.
[7] A. Baronchelli, V. Loreto, A data compression approach to information retrieval, unpublished, http://arxiv.org/abs/cond-mat/0403233.
[8] A.D. Wyner, Shannon lecture, Typical sequences and all that: entropy, pattern matching and data compression, IEEE Information Theory Society Newsletter, 1944.
[9] D. Loewenstern, H. Hirsh, P. Yianilos, M. Noordewieret, DNA sequence classification using compression-based induction, DIMACS Technical Report 95-04, 1995.
[10] O.V. Kukushkina, A.A. Polikarpov, D.V. Khmelev, Using Literal and Grammatical Statistics for Authorship Attribution. Problemy Peredachi Informatsii, 37 (2000) 96–108 (in Russian). Translated in English, in: Problems of Information Transmission, Vol. 37, 2001, pp. 172–184.
[11] D. Benedetto, E. Caglioti, V. Loreto, Language trees and zipping, Phys. Rev. Lett. 88 (2002) 048702–048705.
[12] A. Puglisi, D. Benedetto, E. Caglioti, V. Loreto, A. Vulpiani, Data compression and learning in time sequences analysis, Physica D 180 (2003) 92.
[13] J. Ziv, N. Merhav, A measure of relative entropy between individual sequences with applications to universal classification, IEEE Trans. Inform. Theory 39 (1993) 1280–1292.

[14] C.H. Bennett, P. Gàcs, M. Li, P.M.B. Vitanyi, W. Zurek, Information distance, IEEE Trans. Inform. Theory 44 (1998) 1407–1423;
M. Li, X. Chen, X. Li, B. Ma, P.M.B. Vitanyi, The similarity metric, Proc. 14th ACM-SIAM Symp., to appear in IEEE Trans Information Theory, http://arxiv.org/abs/cs.CC/0111054;
R. Cilibrasi, P.M.B. Vitanyi, Clustering by compression, http://arxiv.org/abs/cs.CV/0312044.
[15] A. Kaltchenko, Algorithms for estimating information distance with application to bioinformatics and linguistics, http://xxx.arxiv.cornell.edu/abs/cs.CC/0404039.
[16] Liberliber homepage: http://www.liberliber.it.
[17] M. Li, X. Chen, J.H. Badger, S. Kwong, P. Kearney, H. Zhang, An information based sequence distance and its application to whole mitochondria genome distance, Bioinformatics 17 (2001) 149–154.
[18] D.T. Pride, R.J. Meinersmann, T.W. Wassenaar, M.J. Blaser, Evolutionary implications of microbial genome tetranucleotide frequency bias, Genome Res. 13 (2003) 145–158.