Language Trees and Zipping

Dario Benedetto,[1,*] Emanuele Caglioti,[1,†] and Vittorio Loreto[2,3,‡]

[1] *"La Sapienza" University, Mathematics Department, Piazzale Aldo Moro 5, 00185 Rome, Italy*
[2] *"La Sapienza" University, Physics Department, Piazzale Aldo Moro 5, 00185 Rome, Italy*
[3] *INFM, Center for Statistical Mechanics and Complexity, Rome, Italy*

(Received 29 August 2001; revised manuscript received 13 September 2001; published 8 January 2002)

In this Letter we present a very general method for extracting information from a generic string of characters, e.g., a text, a DNA sequence, or a time series. Based on data-compression techniques, its key point is the computation of a suitable measure of the remoteness of two bodies of knowledge. We present the implementation of the method to linguistic motivated problems, featuring highly accurate results for language recognition, authorship attribution, and language classification.

DOI: 10.1103/PhysRevLett.88.048702 PACS numbers: 89.70.+c, 01.20.+x, 05.20.–y, 05.45.Tp

Many systems and phenomena in nature are often represented in terms of sequences or strings of characters. In experimental investigations of physical processes, for instance, one typically has access to the system only through a measuring device which produces a time record of a certain observable, i.e., a sequence of data. On the other hand, other systems are intrinsically described by a string of characters, e.g., DNA and protein sequences, language.

When analyzing a string of characters the main question is to extract the information it brings. For a DNA sequence this would correspond to the identification of the subsequences codifying the genes and their specific functions. On the other hand, for a written text one is interested in *understanding* it, i.e., recognizing the language in which the text is written, its author, the subject treated, and eventually the historical background.

The problem being cast in such a way, one would be tempted to approach it from a very interesting point of view: that of information theory [1,2]. In this context the word information acquires a very precise meaning, namely that of the entropy of the string, a measure of the *surprise* the source emitting the sequences can reserve to us.

As is evident, the word information is used with different meanings in different contexts. Suppose now for a while having the ability to measure the entropy of a given sequence (e.g., a text). Is it possible to obtain from this measure the information (in the semantic sense) we were trying to extract from the sequence? This is the question we address in this paper.

In particular, we define in a very general way a concept of remoteness (or similarity) between pairs of sequences based on their relative informational content. We devise, without loss of generality with respect to the nature of the strings of characters, a method to measure this *distance* based on data-compression techniques. The specific question we address is whether this informational *distance* between pairs of sequences is representative of the real semantic difference between the sequences. It turns out that the answer is yes, at least in the framework of the examples on which we have implemented the method. We have chosen for our tests some textual corpora and we have evaluated our method on the basis of the results obtained on some linguistic motivated problems. Is it possible to automatically recognize the language in which a given text is written? Is it possible to automatically guess the author of a given text? Last but not the least, is it possible to identify the subject treated in a text in a way that permits its automatic classification among many other texts in a given corpus? In all the cases the answer is positive as we shall give evidence of in the following.

Before discussing the details of our method let us briefly recall the definition of entropy which is closely related to a very old problem, that of transmitting a message without losing information, i.e., the problem of the efficient encoding [3].

The problem of the optimal coding for a text (or an image or any other kind of information) has been enormously studied in the last century. In particular Shannon [1] discovered that there is a limit to the possibility of encoding a given sequence. This limit is the entropy of the sequence. There are many equivalent definitions of entropy, but probably the best definition in this context is the Chaitin–Kolmogorov entropy [4–7]: the entropy of a string of characters is the length (in bits) of the smallest program which produces as output the string. This definition is really abstract. In particular it is impossible, even in principle, to find such a program. Nevertheless there are algorithms explicitly conceived to approach this theoretical limit. These are the file compressors or zippers. A zipper takes a file and tries to transform it in the shortest possible file. Obviously this is not the best way to encode the file but it represents a good approximation of it. One of the first compression algorithms is the Lempel and Ziv algorithm (LZ77) [8,9] (used for instance by *gzip, zip,* and *Stacker*). It is interesting to briefly recall how it works. The LZ77 algorithm finds duplicated strings in the input data. More precisely it looks for the longest match with the beginning of the lookahead buffer and outputs a pointer to that match given by two numbers: a distance, representing how far back the match starts, and a length, representing the number of matching characters. For example, the match of the sequence $\sigma_1 \ldots \sigma_n$ will be represented by the

pointer (d, n), where d is the distance at which the match starts. The matching sequence will then be encoded with a number of bits equal to $\log_2(d) + \log_2(n)$: i.e., the number of bits necessary to encode d and n. Roughly speaking the average distance between two consecutive $\sigma_1 \dots \sigma_n$ is of the order of the inverse of its occurrence probability. Therefore the zipper will encode more frequent sequences with few bytes and will spend more bytes only for rare sequences. The LZ77 zipper has the following remarkable property: if it encodes a sequence of length L emitted by an ergodic source whose entropy per character is s, then the length of the zipped file divided by the length of the original file tends to s when the length of the text tends to ∞ (see [8,10], and references therein). In other words, it does not encode the file in the best way but it does it better and better as the length of the file increases.

The compression algorithms provide then a powerful tool for the measure of the entropy and the fields of applications are innumerous ranging from theory of dynamical systems [11] to linguistics and genetics [12]. Therefore the first conclusion one can draw is about the possibility of measuring the entropy of a sequence simply by zipping it. In this paper we exploit this kind of algorithm to define a concept of remoteness between pairs of sequences.

An easy way to understand where our definitions come from is to recall the notion of relative entropy whose essence can be easily grasped with the following example. Let us consider two ergodic sources $\mathcal{A}$ and $\mathcal{B}$ emitting sequences of 0 and 1: $\mathcal{A}$ emits a 0 with probability p and 1 with probability $1 - p$, while $\mathcal{B}$ emits 0 with probability q and 1 with probability $1 - q$. As already described, the compression algorithm applied to a sequence emitted by $\mathcal{A}$ will be able to encode the sequence almost optimally, i.e., coding a 0 with $-\log_2 p$ bits and a 1 with $-\log_2(1 - p)$ bits. This optimal coding will not be the optimal one for the sequence emitted by $\mathcal{B}$. In particular the entropy per character of the sequence emitted by $\mathcal{B}$ in the coding optimal for $\mathcal{A}$ will be $-q \log_2 p - (1 - q)\log_2(1 - p)$ while the entropy per character of the sequence emitted by $\mathcal{B}$ in its optimal coding is $-q \log_2 q - (1 - q)\log_2(1 - q)$. The number of bits per character wasted to encode the sequence emitted by $\mathcal{B}$ with the coding optimal for $\mathcal{A}$ is the relative entropy (see Kullback-Leibler [13]) of $\mathcal{A}$ and $\mathcal{B}$, $S_{\mathcal{AB}} = -q \log_2 \frac{p}{q} - (1 - q)\log_2 \frac{1-p}{1-q}$.

There exist several ways to measure the relative entropy (see, for instance, [10,14]). One possibility is of course to follow the recipe described in the previous example: using the optimal coding for a given source to encode the messages of another source. The path we follow is along this stream. In order to define the relative entropy between two sources $\mathcal{A}$ and $\mathcal{B}$ we extract a long sequence A from the source $\mathcal{A}$ and a long sequence B as well as a small sequence b from the source $\mathcal{B}$. We create a new sequence $A + b$ by simply appending b after A. The sequence $A + b$ is now zipped, for example using *gzip,* and the measure of the length of b in the coding optimized for A will be $\Delta_{Ab} = L_{A+b} - L_A$, where L_X indicates the length in bits of the zipped file X. The relative entropy $S_{\mathcal{A}}$ per character between $\mathcal{A}$ and $\mathcal{B}$ will be estimated by

$$S_{\mathcal{AB}} = (\Delta_{Ab} - \Delta_{Bb})/|b|, \tag{1}$$

where $|b|$ is the number of characters of the sequence b and $\Delta_{Bb}/|b| = (L_{B+b} - L_B)/|b|$ is an estimate of the entropy of the source $\mathcal{B}$.

Translated in a linguistic framework, if A and B are texts written in different languages, Δ_{Ab} is a measure of the difficulty for a generic person of mother tongue A to understand the text written in the language B. Let us consider a concrete example where A and B are two texts written for instance in English and Italian. We take a long English text and we append to it an Italian text. The zipper begins reading the file starting from the English text. So after a while it is able to encode optimally the English file. When the Italian part begins, the zipper starts encoding it in a way which is optimal for the English, i.e., it finds most of the matches in the English part. So the first part of the Italian file is encoded with the English code. After a while the zipper "learns" Italian, i.e., it tends progressively to find most of the matches in the Italian part with respect to the English one, and changes its rules. Therefore if the length of the Italian file is "small enough" [15], i.e., if most of the matches occur in the English part, the expression (1) will give a measure of the relative entropy. We have checked this method on sequences for which the relative entropy is known, obtaining an excellent agreement between the theoretical value of the relative entropy and the computed value [15]. The results of our experiments on linguistic corpora turned out to be very robust with respect to large variations on the size of the file b [typically 1–15 kilobytes (kbyte) for a typical size of file A of the order of 32–64 kbyte].

These considerations open the way to many possible applications. Though our method is very general [16], in this paper we focus in particular on sequences of characters representing texts, and we shall discuss in particular two problems of computational linguistics: context recognition and the classification of sequences corpora.

Language recognition.—Suppose we are interested in the automatic recognition of the language in which a given text X is written. The procedure we use considers a collection of long texts (a corpus) in as many different (known) languages as possible: English, French, Italian, Tagalog, ... We simply consider all the possible files obtained appending a portion x of the unknown file X to all the possible other files A_i and we measure the differences $L_{A_i+x} - L_{A_i}$. The file for which this difference is minimal will select the language closest to the one of the X file, or exactly its language, if the collection of languages contained this language. We have considered in particular a corpus of texts in 10 official languages of the European Union (UE) [17]: Danish, Dutch, English, Finnish, French,

German, Italian, Portuguese, Spanish, and Swedish. Each text of the corpus played in turn the role of the text X and all the others the role of the A_i. Using in particular 10 texts per language (giving a total corpus of 100 texts) we have obtained that for any single text the method has recognized the language: this means that for any text X the text A_i for which the difference $L_{A_i+x} - L_{A_i}$ was minimum was a text written in the same language. Moreover it turned out that, ranking for each X all the texts A_i as a function of the difference $L_{A_i+x} - L_{A_i}$, all the texts written in the same language were in the first positions. The recognition of the language works quite well for length of the X file as small as 20 characters.

Authorship attribution.—Suppose in this case we are interested in the automatic recognition of the author of a given text X. We shall consider as before a collection, as large as possible, of texts of several (known) authors all written in the same language of the unknown text and we shall look for the text A_i for which the difference $L_{A_i+x} - L_{A_i}$ is minimum. In order to collect a certain statistics we have performed the experiment using a corpus of 90 different texts [18], using for each run one of the texts in the corpus as unknown text. The results, shown in Table I, feature a rate of success of 93.3%. This rate is the ratio between the number of texts whose author has been recognized (another text of the same author was ranked as first) and the total number of texts considered.

The rate of success increases by considering more refined procedures (performing, for instance, weighted averages over the first m ranked texts of a given text). There are, of course, fluctuations in the success rate for each author and this has to be expected since the writing style is something difficult to grasp and define; moreover, it can vary a lot in the production of a single author.

Classification of sequences.—Suppose we have a collection of texts, for instance a corpus containing several versions of the same text in different languages, and that we are interested in a classification of this corpus.

TABLE I. Authorship attribution: For each author depicted we report the number of different texts considered and two measures of success. Number of successes 1 and 2 are the numbers of times another text of the same author was ranked in the first position or in one of the first two positions, respectively.

Author	No. of texts	No. of successes 1	No. of successes 2
Alighieri	8	8	8
D'Annunzio	4	4	4
Deledda	15	15	15
Fogazzaro	5	4	5
Guicciardini	6	5	6
Macchiavelli	12	12	12
Manzoni	4	3	4
Pirandello	11	11	11
Salgari	11	10	10
Svevo	5	5	5
Verga	9	7	9
Total	90	84	89

One has to face two kinds of problems: the availability of large collections of long texts in many different languages and, related to it, the need of a uniform coding for the characters in different languages. In order to solve the second problem we used, for all the texts, the UNICODE [19] standard coding. In order to have the largest possible corpus of texts in different languages we used "The Universal Declaration of Human Rights" [20], which is considered to be the most often translated document in the world; see [21]. Our method, mutuated by the phylogenetic analysis of biological sequences [22–24], considers the construction of a distance matrix, i.e., a matrix whose elements are the distances between pairs of texts. We define the distance by

$$S_{\mathcal{AB}} = (\Delta_{Ab} - \Delta_{Bb})/\Delta_{Bb} + (\Delta_{Ba} - \Delta_{Aa})/\Delta_{Aa}\,, \tag{2}$$

where A and B are indices running on the corpus elements and the normalization factors are chosen in order to be independent of the coding of the original files. Moreover, since the relative entropy is not a distance in the mathematical sense, we make the matrix elements satisfying the triangular inequality. It is important to remark that a rigorous definition of distance between two bodies of knowledge has been proposed by Li and Vitányi [12]. Starting from the distance matrix one can build a tree representation: phylogenetic trees [24], spanning trees, etc. In our example, we have used the Fitch-Margoliash method [25] of the package PhylIP (phylogeny inference package) [26] which basically constructs a tree by minimizing the net disagreement between the matrix pairwise distances and the distances measured on the tree. Similar results have been obtained with the neighbor algorithm [26]. In Fig. 1 we show the results for over 50 languages widespread on the Euro-Asiatic continent. We can notice that essentially all the main linguistic groups (Ethnologue source [27]) are recognized: Romance, Celtic, Germanic, Ugro-Finnic, Slavic, Baltic, Altaic. On the other hand, one has isolated languages such as the Maltese, which is typically considered an Afro-Asiatic language, and the Basque, which is classified as a non-Indo-European language and whose origins and relationships with other languages are uncertain.

Needless to say, a careful investigation of specific linguistics features is not our purpose. In this framework we are interested only in presenting the potentiality of the method for several disciplines.

In conclusion, we have presented here a general method to recognize and classify automatically sequences of characters. We have discussed in particular the application to textual corpora in several languages. We have shown how a suitable definition of remoteness between texts, based on the concept of relative entropy, allows us to extract from a text much important information: the language in

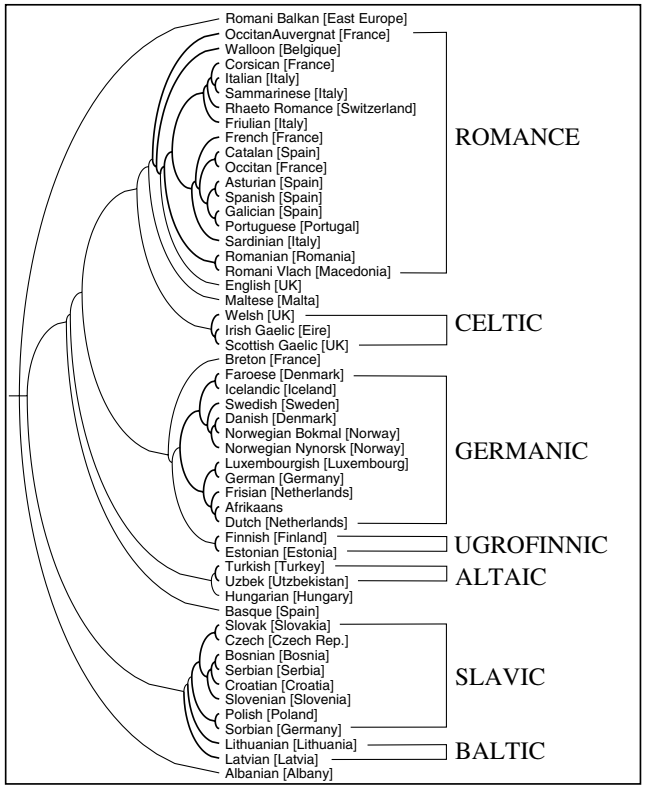

FIG. 1. Language Tree: This figure illustrates the phylogenetic-like tree constructed on the basis of more than 50 different versions of "The Universal Declaration of Human Rights." The tree is obtained using the Fitch-Margoliash method applied to a distance matrix whose elements are computed in terms of the relative entropy between pairs of texts. The tree features essentially all the main linguistic groups of the Euro-Asiatic continent (Romance, Celtic, Germanic, Ugro-Finnic, Slavic, Baltic, Altaic), as well as a few isolated languages such as the Maltese, typically considered an Afro-Asiatic language, and the Basque, classified as a non-Indo-European language, and whose origins and relationships with other languages are uncertain. Notice that the tree is unrooted, i.e., it does not require any hypothesis about common ancestors for the languages. What is important is the relative positions between pairs of languages. The branch lengths do not correspond to the actual distances in the distance matrix.

which it is written, the subject treated as well as its author; on the other hand, the method allows us to classify sets of sequences (a corpus) on the basis of the relative distances among the elements of the corpus itself and organize them in a hierarchical structure (graph, tree, etc.). The method is highly versatile and general. It applies to any kind of corpora of character strings independently of the type of coding behind them: time sequences, language, genetic sequences (DNA, proteins, etc.). It does not require any *a priori* knowledge of the corpus under investigation (alphabet, grammar, syntax) nor about its statistics. These features are potentially very important for fields where the human intuition can fail: DNA and protein sequences, geological time series, stock market data, medical monitoring, etc.

The authors are grateful to Piero Cammarano, Giuseppe Di Carlo, and Anna Lo Piano for many enlightening discussions. This work has been partially supported by the European Network-Fractals under Contract No. FMRXCT980183, GNFM (INDAM).

*Electronic address: benedetto@mat.uniroma1.it
†Electronic address: caglioti@mat.uniroma1.it
‡Electronic address: loreto@roma1.infn.it

[1] C. E. Shannon, Bell Syst. Tech. J. **27**, 379 (1948); **27**, 623 (1948).
[2] *Complexity, Entropy and Physics of Information,* edited by W. H. Zurek (Addison-Wesley, Redwood City, CA, 1990).
[3] D. Welsh, *Codes and Cryptography* (Clarendon Press, Oxford, 1989).
[4] A. N. Kolmogorov, Probl. Inf. Transm. **1**, 1 (1965).
[5] G. J. Chaitin, J. Assoc. Comput. Mach. **13**, 547 (1966).
[6] G. J. Chaitin, *Information Randomness and Incompleteness* (World Scientific, Singapore, 1990), 2nd ed.
[7] R. J. Solomonov, Inf. Control **7**, 1 (1964); **7**, 224 (1964).
[8] A. Lempel and J. Ziv, IEEE Trans. Inf. Theory **23**, 337–343 (1977).
[9] J. Ziv and A. Lempel, IEEE Trans. Inf. Theory **24**, 530 (1978).
[10] A. D. Wyner, "Typical Sequences and All That: Entropy, Pattern Matching and Data Compression" (1994 Shannon Lecture), A.D. IEEE Information Theory Society Newsletter, July 1995.
[11] F. Argenti *et al.,* "Information and Dynamical Systems: A Concrete Measurement on Sporadic Dynamics" (to be published).
[12] M. Li and P. Vitányi, *An Introduction to Kolmogorov Complexity and Its Applications* (Springer, New York, 1997), 2nd ed.
[13] S. Kullback and R. A. Leibler, Ann. Math. Stat. **22**, 79–86 (1951).
[14] N. Merhav and J. Ziv, IEEE Trans. Inf. Theory **39**, 1280 (1993).
[15] D. Benedetto, E. Caglioti, and V. Loreto (to be published).
[16] Patent pending CCIAA Roma N. RM2001A000399.
[17] europa.eu.int
[18] www.liberliber.it
[19] See for details: www.unicode.org
[20] www.unhchr.ch/udhr/navigate/alpha.htm
[21] *Guinness World Records 2002* (Guinness World Records, Ltd., Enfield, United Kingdom, 2001).
[22] L. L. Cavalli-Sforza and A. W. F. Edwards, Evolution **32**, 550–570 (1967); Am. J. Human Genet. **19**, 233–257 (1967).
[23] J. S. Farris, *Advances in Cladistics: Proceedings of the First Meeting of the Willi Hennig Society,* edited by V. A. Funk and D. R. Brooks (New York Botanical Garden, Bronx, New York, 1981), pp. 3–23.
[24] J. Felsenstein, Evolution **38**, 16–24 (1984).
[25] W. M. Fitch and E. Margoliash, Science **155**, 279–284 (1967).
[26] J. Felsenstein, Cladistics **5**, 164–166 (1989).
[27] www.sil.org/ethnologue