PHYSICAL REVIEW E **79**, 035102(R) (2009)

Level statistics of words: Finding keywords in literary texts and symbolic sequences

P. Carpena,[1] P. Bernaola-Galván,[1] M. Hackenberg,[2] A. V. Coronado,[1] and J. L. Oliver[3]
[1]*Departamento de Física Aplicada II, Universidad de Málaga, 29071 Málaga, Spain*
[2]*Bioinformatics Group, CIC bioGUNE, Technology Park of Bizkaia, 48160 Derio, Bizkaia, Spain*
[3]*Departamento de Genética, Universidad de Granada, 18071 Granada, Spain*
(Received 21 May 2008; published 10 March 2009)

Using a generalization of the level statistics analysis of quantum disordered systems, we present an approach able to extract automatically keywords in literary texts. Our approach takes into account not only the frequencies of the words present in the text but also their spatial distribution along the text, and is based on the fact that relevant words are significantly clustered (i.e., they self-attract each other), while irrelevant words are distributed randomly in the text. Since a reference *corpus* is not needed, our approach is especially suitable for single documents for which no *a priori* information is available. In addition, we show that our method works also in generic symbolic sequences (continuous texts without spaces), thus suggesting its general applicability.

DOI: 10.1103/PhysRevE.79.035102 PACS number(s): 89.20.−a, 89.65.−s, 89.75.Fb, 05.50.+q

Statistical keyword extraction is a critical step in information science, with multiple applications in text-mining and information-retrieval systems [1]. Since Luhn [2] proposed the analysis of frequency occurrences of words in the text as a method for keyword extraction, many refinements have been developed. With a few exceptions [3], the basic principle for keyword extraction is the comparison to a *corpus* of documents taken as a reference. For a collection of documents, modern term-weighting schemes use the frequency of a term in a document and the proportion of documents containing that term [4]. Following a different approach, the probabilistic model of information retrieval related the significance of a term to its frequency fluctuations between documents [5–7]. The frequency analysis approach to detect keywords seems to work properly in this context.

However, a more general approach should try to detect keywords in a single text without knowing *a priori* the subject of the text, i.e., without using a corpus of reference. The applications of such an algorithm are clear: internet searches, data mining, automatic classification of documents, etc. In this case, the information provided by the frequency of a word is not very useful, since there are no more texts to compare. In addition, such frequency analysis is of little use in a single document for two main reasons: (i) Two words with very different relevance in the text can have a similar frequency (see Fig. 1). (ii) A randomization of the text preserves the frequency values but destroys the information, which must be also stored in the ordering of the words, and not only in the words themselves. Thus, to detect keywords, we propose the use of the spatial distribution of the words along the text and not only their frequencies, in order to take into account the structure of the text as well as its composition.

Inspired by the level statistics of quantum-disordered systems following the random matrix theory [8], Ortuño *et al.* [9] have shown that the spatial distribution of a relevant word in a text is very different from that corresponding to a nonrelevant word. In this approach, any of the occurrences of a particular word is considered as an "energy level" e_i within an "energy spectrum" formed by all the occurrences of the analyzed word within the text. The value of any energy level e_i is given simply by the position of the analyzed word in the text. For example, in the sentence "A great scientist must be a good teacher and a good researcher" the spectrum corresponding to the word "a" is formed by three energy levels (1,6,10). Figure 1 shows an example of a real book.

Following the physics analogy, the nearest-neighbor spacing distribution $P(d)$ was used in [9] to characterize the spatial distribution of a particular word, and to show the relationship between word clustering and word semantic meaning. $P(d)$ is obtained as the normalized histogram of the sets of distances (or spacings) $(d_1, d_2, \ldots, d_n)$ between consecutive occurrences of a word, with $d_i = e_{i+1} - e_i$. As seen in Fig. 1, a nonrelevant word (as "but") is placed at random along the text, while a relevant word (as "Quixote") appears in the text forming clusters, and this difference is reflected in their corresponding $P(d)$ distributions. In the case of a relevant word the energy levels attract each other, while for a nonrelevant word, the energy levels are uncorrelated and therefore distributed at random, so the higher the relevance of a word, the larger the clustering (the attraction) and the larger the deviation of $P(d)$ from the random expectation. The connection between word attraction (clustering) and relevance comes from the fact that a relevant word is usually the main subject on local contexts, and therefore it appears

FIG. 1. Spectra of the words "Quixote" and "but" obtained in the first 50 000 words of the book *Don Quixote*, by Miguel de Cervantes. Both words have a similar frequency in the whole text (around 2150), and also in the part shown in the figure.

more often in some areas and less frequently in others, giving rise to clusters.

We present here an approach to the problem of keyword detection with does not need a *corpus* and which is based on the principle that real keywords are clustered in the text (as in [9]), but in addition we introduce the idea that the clustering must be statistically significant, i.e., not due to statistical fluctuations. This is of fundamental importance when analyzing any text, but is critical when analyzing short documents (articles, etc.) where fluctuations are important since all the words present a small frequency. As the statistical fluctuations depend on the frequency of the corresponding word (see below), our approach combines both the information provided by the clustering of the word (the spatial structure along the text) and by its frequency. Furthermore, we extend our approach to general symbolic sequences, where word boundaries are not known. In particular, we model a generic symbolic sequence (i.e., a chain of "letters" from a certain "alphabet") by an ordinary text without blank spaces between words. As we know *a priori* the correct hidden keywords this allows us to test our method. We show that the clustering (attraction) experienced by keywords is still observable in such a text, and therefore that real keywords can be detected even when the "words" of the text are not known.

To quantify the clustering (and thus the relevance) of a word using a single parameter instead of the whole distribution $P(d)$, in [9] the parameter σ was used defined as $\sigma \equiv s/\langle d\rangle$, with $\langle d\rangle$ being the average distance and $s = \sqrt{\langle d^2\rangle - \langle d\rangle^2}$ the standard deviation of $P(d)$. For a particular word, σ is the standard deviation of its normalized set of distances $\{d_1/\langle d\rangle, d_2/\langle d\rangle, \ldots, d_n/\langle d\rangle\}$, i.e., distances given in units of the mean distance, which allows the direct comparison of the σ values obtained for words with different frequency. The use of σ to characterize $P(d)$ is common in the analysis of energy levels of quantum disordered systems [10]. For these systems, when the energy levels are uncorrelated and behave randomly, the corresponding $P(d)$ is the Poisson distribution [8], $P(d)=e^{-d}$, for which $\sigma=1$. Thus in [9] the value expected for a nonrelevant word without clustering and distributed randomly in a text was $\sigma=1$, and the larger σ, the larger the clustering (and the relevance) of the corresponding word. This approach proved to be fruitful and later works used it to test keywords detection [11].

However, the cluster-free (random) distribution $P(d)$ is Poissonian only for a continuous distance distribution, which is valid for the energy levels, but not for the words, where the distances are integers. The discrete counterpart of the Poisson distribution is the geometric one:

$$P_{\mathrm{geo}}(d) = p(1-p)^{d-1}, \tag{1}$$

where $p=n/N$ is the probability of the word within the text, n being the counts of the corresponding word and N the total number of words in the text. $P_{\mathrm{geo}}(d)$ is expected for a word placed at random in a text. Examples of $P(d)$ for words distributed according to more complex models than the random one can be found, for example, in [12,13], but we have observed that the geometric distribution is a very good model to describe the behavior of unrelevant words, and therefore

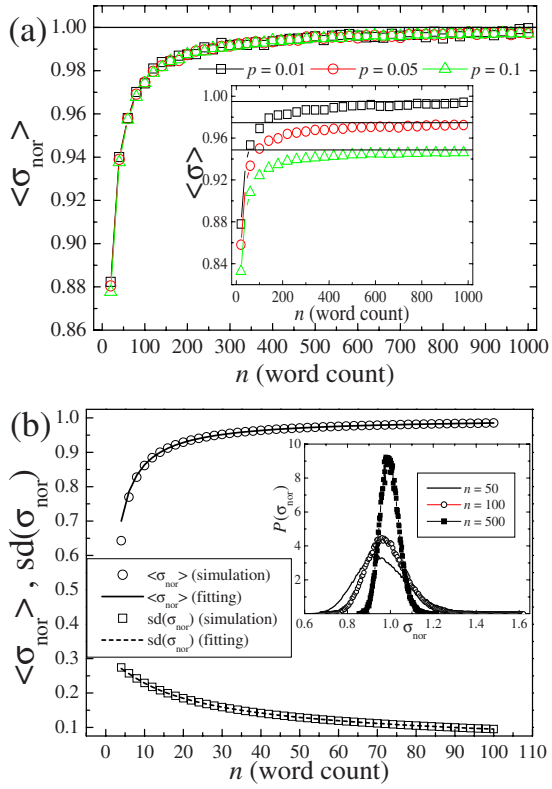

FIG. 2. (Color online) (a) $\langle\sigma_{\mathrm{nor}}\rangle$ as a function of the word count n for words with different p in a random text. The horizontal line is the value $\sigma_{\mathrm{nor}}=1$. Inset: the same, but for σ instead of σ_{nor}. The horizontal lines are the expected values $\sqrt{1-p}$. (b) The mean $\langle\sigma_{\mathrm{nor}}\rangle$ and the standard deviation $\mathrm{sd}(\sigma_{\mathrm{nor}})$ of the probability distribution $P(\sigma_{\mathrm{nor}})$ as a function of n obtained by simulation of random texts. The solid lines correspond to fittings according to Eq. (3). Inset: $P(\sigma_{\mathrm{nor}})$ for three different n.

we use it as our null hypothesis. For the geometric case, $\sigma_{geo}=\sqrt{1-p}$ since $s=\sqrt{1-p}/p$ and $\langle d\rangle = 1/p$, and the continuum case ($\sigma=1$) is recovered when $p\to 0$. Thus, in the discrete case, words with different p randomly placed in a text would give a different clustering level σ [see Fig. 2(a), inset]. To eliminate this effect, we define the clustering measure σ_{nor} as

$$\sigma_{\mathrm{nor}} = \frac{\sigma}{\sigma_{geo}} = \frac{\sigma}{\sqrt{1-p}}. \tag{2}$$

To show that this correction is effective, in Fig. 2(a) we plot the behavior of the average value of σ_{nor} for words with different p in a simulation of random texts [14]: all the curves collapse into a single one independently on p, showing that the normalization works, and the expected value $\sigma_{\mathrm{nor}}=1$ is recovered in all cases (for large n).

However, in σ_{nor} the influence of the word count n is not considered, although it can be of critical importance. The mean value $\langle\sigma_{\mathrm{nor}}\rangle$ presents clear finite size effects (is biased) [Fig. 2(a)]. The strong n dependence also appears in the standard deviation of the distribution of σ_{nor} values [$\mathrm{sd}(\sigma_{\mathrm{nor}})$] and in the whole distribution $P(\sigma_{\mathrm{nor}})$ itself [Fig. 2(b)]. As expected, for small n the distribution $P(\sigma_{\mathrm{nor}})$ is wide, and the

TABLE I. The first 20 "words" extracted from the book *Relativity: The Special and General Theory*, by A. Einstein, with spaces and punctuation marks removed.

Word	Counts	σ_{nor}	C
energy	23	4.29	19.10
theuniverse	20	3.84	15.76
erical	26	3.25	13.74
project	35	2.73	11.85
alongthe	17	2.92	10.28
econtinuum	23	2.70	10.04
thegravitationalfield	27	2.60	10.01
sphere	16	2.8	9.79
electron	13	2.92	9.54
geometry	31	2.45	9.54
theprincipleofrelativity	33	2.41	9.48
specific	11	2.91	9.11
theembankment	40	2.25	9.09
square	28	2.41	8.92
thetheoryofrelativity	32	2.31	8.78
velocityv	17	2.60	8.63
referencebody	56	2.01	8.50
materialpoint	12	2.69	8.29
thelorentztransformation	33	2.22	8.26
fourdimensional	26	2.33	8.25

probability of having by chance large σ_{nor} values is not negligible. As n increases, $P(\sigma_{\mathrm{nor}})$ becomes narrower and consists essentially of a Gaussian peak centered at $\sigma_{\mathrm{nor}}=1$: now, the probability of having by chance large σ_{nor} values is very small. As a consequence, this strong n dependence can be crucial: since the statistical fluctuations [as measured by $\mathrm{sd}(\sigma_{\mathrm{nor}})$] are much larger for small n, it is possible to obtain a larger σ_{nor} for a rare word placed at random in a text than for a more frequent real keyword. The rare random word would be misidentified as a keyword.

To solve this problem we propose a new relevance measure which takes into account not only the clustering of the word measured by σ_{nor}, but also its statistical significance given the word counts n. To achieve this, we obtained first by extensive simulation of random texts [14] the n dependence (bias) of the mean value $\langle\sigma_{\mathrm{nor}}\rangle$ and the standard deviation $\mathrm{sd}(\sigma_{\mathrm{nor}})$ of the distribution $P(\sigma_{\mathrm{nor}})$, which are shown in Fig. 2(b). Both functions are very well fitted in the whole n range by

$$\langle\sigma_{\mathrm{nor}}\rangle = \frac{2n-1}{2n+2}, \quad \mathrm{sd}(\sigma_{\mathrm{nor}}) = \frac{1}{\sqrt{n}(1+2.8n^{-0.865})}. \tag{3}$$

Note how for large n, $\langle\sigma_{\mathrm{nor}}\rangle \to 1$ and $\mathrm{sd}(\sigma_{\mathrm{nor}}) \to 1/\sqrt{n}$, in agreement with the central limit theorem.

As $P(\sigma_{\mathrm{nor}})$ tends to be Gaussian, we can design an appropriate relevance measure C: for a word with n counts and a given σ_{nor} value, we define the measure C as

$$C(\sigma_{\mathrm{nor}},n) \equiv \frac{\sigma_{\mathrm{nor}} - \langle\sigma_{\mathrm{nor}}\rangle(n)}{\mathrm{sd}(\sigma_{\mathrm{nor}})(n)}, \tag{4}$$

i.e., C measures the deviation of σ_{nor} with respect to the expected value in a random text $[\langle\sigma_{\mathrm{nor}}\rangle(n)]$ in units of the expected standard deviation $[\mathrm{sd}(\sigma_{\mathrm{nor}})(n)]$. Thus C is a Z-score measure which depends on the frequency n of the word considered, and combines the clustering of a word and its frequency. To calculate C we use the numerical fittings of Eq. (3). $C=0$ indicates that the word appears at random, $C>0$ that the word is clustered, and $C<0$ that the word repels itself. In addition, two words with the same C value can have different clustering (different σ_{nor} value), but the same statistical significance.

We used systematically C to analyze a large collection of texts [15] (novels, poetry, scientific books). C can be used in two ways: (i) to rank the words according to their C values and (ii) to rank the words according to their σ_{nor} values but only for words with a C value larger than a threshold value C_0, which fixes the statistical significance considered. Both approaches work extremely well for many texts in different languages [16].

The Origin of Species by Means of Natural Selection is a good example to understand the effect of C: using σ_{nor}, for the very relevant word "species" ($n=1922$) we have $\sigma_{\mathrm{nor}}=1.905$. In the σ_{nor}-ranking "species" appears in the 505th place! Nevertheless, when using the C measure we find for this word $C=39.97$, and in the C ranking it is in the 5th place (after "sterility," "hybrids," "varieties," and "instincts").

Next, we would like to extract keywords from symbolic sequences, considered as a continuous chain of "letters" without "spaces" separating them. Previous attempts in this direction [17] were based on word frequencies and not on the spatial structure of the text. Our underlying idea is that even in a symbolic sequence the spatial distribution of relevant "words" should be different of the irrelevant ones, and the clustering approach can provide useful results.

To model generic symbolic sequences, we use standard, literary texts in which all the spaces, punctuation marks, etc., have been removed, thus producing a continuous chain of letters drawn from the alphabet $\{a,b,\ldots,z,0,1,\ldots,9\}$ [18]. Since we know *a priori* the real keywords hidden in such texts, this may be a good benchmark for our method. Our approach works as follows: as true "words" are unknown, we calculate the C measure [19] for all possible ℓ-letter words, where ℓ is a small integer ($\ell=2-35$). For each ℓ, we rank the ℓ words by their C values. As the number of different ℓ words is immense (x^{ℓ}, with x the number of letters of the alphabet), keyword detection is a daunting task. Note that, for a given ℓ, any word contains many other words of smaller ℓ and is also part of words with larger ℓ. In this way, the putative words can be viewed as a direct acyclic graph (DAG) [20]. DAGs are hierarchical treelike structures where each child node can have various parent nodes. Parent nodes are general words (shorter words, small ℓ) while child nodes are more specific words (larger words, large ℓ). For a given ℓ, each ℓ word has two $(\ell-1)$ parents (for example, the word "energy" has "energ" and "nergy") and $2x$ $(\ell+1)$ children (like "eenergy," "energya," etc.). As expected, we observed

that words with semantic meaning and their parents are strongly clustered, while irrelevant and common words are randomly distributed.

For keyword extraction we use two principles: (i) for any ℓ, we apply a threshold for C to remove irrelevant words which is taken as a percentile of the C distribution, usually a p value $\leqslant 0.05$). And (ii) we explore the "lineages" of all words (from short, "general" ℓ words to larger, "specialized" ones) to extract just the words with semantic meaning and not any of their parents which might be also highly clustered. The lineage of a word can easily be established by iteratively detecting the child word with the highest C value. The result of such an algorithm [21] for a famous book without spaces and punctuation marks can be seen in Table I, and for other books in [16]. It is remarkable that this algorithm, based on the spatial attraction of the relevant words, when applied to a text without spaces is not only able to detect real hidden keywords, but also a combination of words or whole sentences with plenty of meaning for the text considered, thus supporting the validity of our approach.

In conclusion, our algorithm has proven to work properly in a generic symbolic sequence, being thus potentially useful when analyzing other specific symbolic sequences of interest, as for example, spoken language, where only sentences can be preceded and followed by silence but not the individual words, or DNA sequences, where commaless codes are the rule.

We thank the Spanish Junta de Andalucía (Grant Nos. P07-FQM3163 and P06-FQM1858) and the Spanish Government (Grant No. BIO2008-01353) for financial support.

[1] *WordNet: An Electronic Lexical Database*, edited by C. Fellbaum (MIT Press, Cambridge, MA, 1998); E. Frank *et al.*, *Proceedings of the 16th International Joint Conference on Artificial Intelligence*, 1999, p. 668; L. van der Plas *et al.*, in *Proceedings of the 4th International Conference on Language Resources and Evaluation*, edited by M. T. Lino, M. F. Xavier, F. Ferreira, R. Costa, and R. Silva, European Language Resource Association, 2004, p. 2205; K Cohen and L. Hunter, PLOS Comput. Biol. **4**, e20 (2008).

[2] H. P. Luhn, IBM J. Res. Dev. **2**, 157 (1958).

[3] Y. Matsuo and M. Ishizuka, Int. J. Artif. Intell. **13**, 157 (2004); G. Palshikar, in *Proceedings of the Second International Conference on Pattern Recognition and Machine Intelligence (PReMI07)*, Vol. 4815 of Lecture Notes on Computer Science, edited by A. Ghosh, R. K. De, and S. K. Pal (Springer-Verlag, Berlin, 2007), p. 503.

[4] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval* (McGraw-Hill, New York, 1983); K. Sparck Jones, J. Document. **28**, 11 (1972); S. E. Robertson *et al.*, in *Overview of the Third Text REtrieval Conference (TREC-3)*, NIST SP 500–225, edited by D. K. Harman (National Institute of Standards and Technology, Gaithersburg, MD, 1995), pp. 109–126.

[5] A. Bookstein and D. R. Swanson, J. Am. Soc. Inf. Sci. **25**, 312 (1974); **26**, 45 (1975).

[6] S. P. Harter, J. Am. Soc. Inf. Sci. **26**, 197 (1975).

[7] A. Berger and J. Lafferty, *Proc. ACM SIGIR99*, 1999, p. 222; J. Ponte and W. Croft, *Proc. ACM SIGIR98*, 1998, p. 275.

[8] T. A. Brody *et al.*, Rev. Mod. Phys. **53**, 385 (1981); M. L. Mehta, *Random Matrices* (Academic Press, New York, 1991).

[9] M. Ortuño *et al.*, Europhys. Lett. **57**, 759 (2002).

[10] P. Carpena, P. Bernaola-Galvan, and P. C. Ivanov, Phys. Rev. Lett. **93**, 176804 (2004).

[11] H. D. Zhou and G. W. Slater, Physica A **329**, 309 (2003); M. J. Berryman, A. Allison, and D. Abbott, Fluct. Noise Lett. **3**, L1 (2003).

[12] V. T. Stefanov, J. Appl. Probab. **40**, 881 (2003).

[13] S. Robin and J. J. Daudin, Ann. Inst. Stat. Math. **53**, 895 (2001).

[14] We simulate random texts as random binary sequences 0100101000001…. The symbol "1" appears with probability p and models a word in a text, and the symbol "0" accounts for the rest of the words with probability $1-p$, so in a realistic case p is very small.

[15] All the texts analyzed have been downloaded from the Project Gutenberg web page: www.gutenberg.org

[16] http://bioinfo2.ugr.es/TextKeywords

[17] H. J. Bussemaker, H. Li, and E. D. Siggia, Proc. Natl. Acad. Sci. U.S.A. **97**, 10096 (2000).

[18] We only use lowercase letters to reduce the alphabet.

[19] Now, the nearest neighbors distances are measured in letters, not in words, since we advance letter by letter exploring the text. Thus we ignore the "reading frame," which would be relevant only in texts with fixed-length words consecutively placed, as the words of length 3 (codons) in the coding regions of DNA. In addition, we consider only nonoverlapping occurrences of a word. Although the overlapping is not very likely in text without spaces, it can happen in other symbolic sequences, such as DNA, and the expected $P(d)$ is different for overlapping or nonoverlapping randomly placed words (see S. Robin, F. Rodolphe, and S. Schbath, *DNA, Words and Models* (Cambridge University Press, Cambridge, England, 2005).

[20] T. H. Cormen *et al.*, *Introduction to Algorithms* (The MIT Press, Cambridge, MA, 1990), pp. 485–488.

[21] The algorithm proceeds as follows: we consider an initial ℓ_0, and we start with a given ℓ_0 word for which $C > C_0$, where C_0 corresponds to a p value $= 0.05$. We then find the child $(\ell_0 + 1)$ word with the highest C value, and proceed the same in the following generations: we find the successive child $(\ell_0 + i)$ word with the highest C value; $i = 1, 2, \ldots$. This process stops at a previously chosen maximal word length $\ell_{\max}$. Finally, we choose as the representative of the lineage the longest word for which $C > C_0$ and define it as the extracted semantic unit. We repeat this algorithm for all the ℓ_0 words with $C > C_0$, and we repeat also all the processes by changing the initial ℓ_0 value. Finally, we remove the remaining redundancies (repeated words or semantic units) due to explore all lineages from different initial ℓ_0.