

Spectral methods cluster words of the same class in a syntactic dependency network

Ramon Ferrer i Cancho,¹ Andrea Capocci,² and Guido Caldarelli^{1,2}¹*INFN UdR Roma1, Dip. di Fisica, Università “La Sapienza”. Piazzale A. Moro 5, 00185 Roma, Italy.*²*Centro Studi e Ricerche e Museo della Fisica, “E. Fermi”, Compendio Viminale, Roma, Italy.*

(Dated: August 15, 2005)

We analyze here a particular kind of linguistic network where vertices represent words and edges stand for syntactic relationships between words. The statistical properties of these networks have been recently studied and various features such as the small-world phenomenon and a scale-free distribution of degrees have been found. Our work focuses on four classes of words: verbs, nouns, adverbs and adjectives. Here, we use spectral methods sorting vertices. We show that the ordering clusters words of the same class. For nouns and verbs, the cluster size distribution clearly follows a power-law distribution that cannot be explained by a null hypothesis. Long-range correlations are found between vertices in the ordering provided by the spectral method. The findings support the use of spectral methods for detecting community structure.

PACS numbers: 89.75.-k, 89.20.-a

Keywords: complex networks, communities, word classes

I. INTRODUCTION

A great amount of efforts has been recently devoted to the study of complex networks [1, 2]. The topology of systems as different as the Internet [3, 4], the World Wide Web [5, 6, 7], biological [8] and social systems [9] have been found to share similar statistical properties. Many networks display both a distribution of distances (defined as the minimum number of edges between two vertices) peaked around a small characteristic value (i.e. *small world* [10] effect) and the degree (defined as the number of links per vertex) is distributed according to a power law function. Driven by the widespread presence of these common features, much work has been done in order to understand the basic mechanisms underlying such universality [1, 11, 12, 13]. Instead, we focus on a further characterization of networks using spectral methods that have been used for detecting communities in complex networks [14, 15, 16, 17]. Community detection techniques represent now a flourishing, inter-disciplinary field [14, 18, 19, 20, 21, 22, 23, 24]. The aim of the present paper is to show how spectral methods [14, 15, 16, 17] cluster four classes of content words: verbs, nouns, adverbs and adjectives in a Syntactic Dependency Network [25] (SDN), which is a kind of linguistic network. The topology of various instances of linguistic networks has recently been under consideration by several studies. Examples include thesaurus networks based on the Roget’s thesaurus [12, 26, 27] and networks based on Merriam-Webster’s thesaurus [1], WordNet [26, 28], word association networks [14, 26], word co-occurrence networks [29], and SDNs [25]. The latter, as said above, will be the target of the present article.

SDNs are constructed by collecting the syntactic dependency links from a corpus, i.e. a set of sentences. Syntactic links are defined according to the dependency grammar formalism [30], a special case of a broad family of grammatical formalisms [31, 32]. Dependency grammar assumes that the syntactic structure of sentences

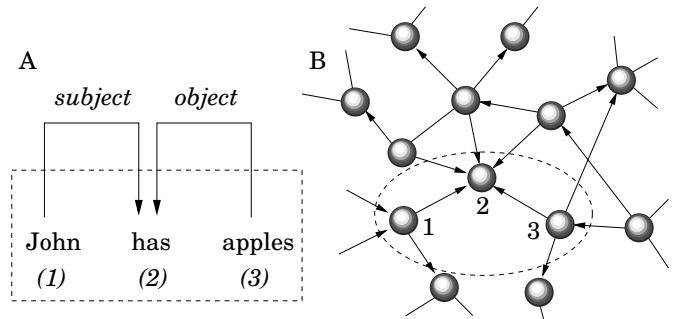


FIG. 1: A) The syntactic structure of a simple sentence. Words are the vertices and the syntactic dependencies are the edges of the graph. The proper noun ‘John’ and the verb ‘has’ are syntactically dependent in this sample sentence. ‘John’ is *modifier* of the verb ‘has’, which is its *head*. Similarly, the action of ‘has’ is modified by its object ‘apples’. Here we assume the graph oriented with edges pointing from a *modifier* to its *head*. B) Mapping the syntactic dependency structure of the sentence into a global syntactic dependency network.

consists of vertices (words) and word pairwise connections (syntactic dependencies). In this approximation, a dependency link connects a pair of words representing an edge of the graph. In the simple sentence “John has apples”, “John” is linked to “has” and “apples” is linked to “has”. As explained in Fig. 1, we can assign a direction to this edges. “John” has the function to modify the meaning of the word “has” so “John” plays the role of *modifier* and the word “has” is referred as *head*. Most of edges are then directed and we assume that arrows go from the modifier to its head (other conventions may make the opposite choice). In some cases, such as in coordination, there is no clear direction [33]. But since such cases are rather uncommon, we will assume that every link has a definite direction, by assigning an arbitrary direction to the undirected cases.

We define a SDN as a set of n words labeled with natu-

rals from 1 to n . Syntactic dependencies are specified by the adjacency matrix $A = \{a_{ij}\}$. $a_{ij} = 1$ if the i -th word points to the j -th word, and $a_{uj} = 0$ otherwise. In fact, $a_{ij} = 1$ are connected if the i -th has pointed to the j -th word at least once in a sentence from a corpus. The syntactic dependency structure of a sentence can be seen as a subset of vertices and links contained in a global network (Fig. 1 B). Syntactic dependencies between words in a sentence tend to be local and the distance between syntactically linked words in a sentence decays exponentially [34, 35]. The organization of the rest of the papers can be summarized as follows. Section II gives a brief account of community detection techniques, with a special emphasis on spectral methods and the way they will be applied here. Section III describes the sources of the data and the statistical measures that will be used to show how the spectral methods cluster words of the same class. Section IV presents the results. A discussion of the findings and some conclusions are reported in section V.

II. THE COMMUNITY DETECTION TECHNIQUE

Various techniques for detecting community structure have been proposed recently [14, 18, 19, 20, 21, 22, 23, 24]. Some of them [19, 36] use the same arguments underlying the algorithm introduced by Newman and Girvan [20] (NG-algorithm), and they focus on the value of the edge betweenness. This quantity measures the fraction of all shortest paths passing over a given link, or, according to an alternative definition, the probability that a random walk on the network runs through that link. By removing edges with large betweenness, one splits the whole network (step by step) into connected components. The process goes on until the whole graph is decomposed in communities consisting of one single node each. The method is very efficient whenever some clues on the community structure is at hand. Otherwise, it does not give an indication of the resolution of the clustering. Therefore it needs extra information in input (like the expected number of clusters). Furthermore its outcome is independent on how sharp the partitioning of the graph is.

To overcome such problems we adopted a different approach based on the spectral analysis [14]. That approach conjugates the power of spectral analysis with the caution needed to reveal an underlying structure when there is no clear cut partitioning, as is in the case of the SDN considered.

Spectral methods are based on the analysis of simple functions of the adjacency matrix A [15, 16, 17]. In particular, the most widely investigated function of A are the Laplacian matrix $L = K - A$, the Normal matrix $N = K^{-1}A$ and the composition AA^T , particularly useful when dealing with oriented graphs. In the quantities defined above, we have assumed that K is the diagonal matrix with elements $k_{ii} = \sum_{j=1}^n a_{ij}$ and n is the number of nodes in the network. In most approaches concerning

undirected networks, A is assumed to be symmetric, in contrast with the present case, which explains our emphasis on the matrix AA^T .

Just to familiarize the reader with the concept of the connection between spectral analysis and communities, let us consider the simplest case among those described above, the matrix N . Elements in a row can be interpreted as probabilities that a walker moves from a given vertex to the other ones, since those elements sum up to one. By such a probabilistic approach, it is evident that the largest eigenvalue of the matrix N is always equal to one. This eigenvalue is associated to a trivial constant eigenvector, due to row normalization. In a network with an apparent cluster structure, N has also a certain number $m - 1$ of eigenvalues close to one, where m is the number of well defined communities, the remaining eigenvalues lying a gap away from one. We denote by x_i the position of the i -th vertex after sorting vertices by the value of its corresponding component in one eigenvector. The eigenvectors associated to these largest $m - 1$ eigenvalues have a characteristic structure too: the components corresponding to nodes within the same cluster are assigned very similar x_i values so that, as long as the partition is sufficiently sharp, the profile of each eigenvector, sorted by components, is step-like. The number of steps in the profile gives again the number m of communities [14]. A similar information is encoded in the non-negative definite Laplacian matrix, where the eigenvalues close to zero are associated to clusters. While one can easily show that these spectral properties are associated with the clustering pattern in the Normal matrix case, this is less evident for the other cases. Nevertheless, it has been shown [16] that the spectral structure of the AA^T helps in detecting sets of highly mutually connected nodes in directed networks, thereby indicating the presence of communities in the network such as the World Wide Web. We will use the same empirical evidence here for the analysis of our word dataset.

As explained in [14], solving the eigenvalue problem is equivalent to minimization of a suitable function of the x_i s under a suitable constraint. The absolute minimum corresponds to the trivial eigenvector, which is constant. The remaining stationary points correspond to eigenvectors where components associated to well connected nodes assume similar values.

For the present aim, it is enough to mention that (as for general real networks) the typical eigenvector profiles in our SDN are not step-like, but rather resemble a continuous curve. Nevertheless, the method still applies. In fact, we will see that components corresponding to nodes belonging to the same class (or equivalently, that do not belong to a certain class) are still strongly correlated and take, in each eigenvector, similar values among themselves. Thus, a natural way to identify communities in an automatic manner is to measure the correlation

$$r_{ij} = \frac{\langle x_i x_j \rangle - \langle x_i \rangle \langle x_j \rangle}{[(\langle x_i^2 \rangle - \langle x_i \rangle^2)(\langle x_j^2 \rangle - \langle x_j \rangle^2)]^{\frac{1}{2}}}, \quad (1)$$

where the average $\langle \cdot \rangle$ is computed over the first few non-trivial eigenvectors. The quantity r_{ij} measures the community closeness between the vertices i and j . Though the performance may be improved by averaging over a larger number of eigenvectors, with increased computational effort, we found that indeed a small number of eigenvectors suffices to measure the correlation between two vertices, which is positively correlated with the chance that the two networks are strongly correlated [14].

The spectral method is used in a different manner here. The class each node belongs to is known in advance; we check, then, whether the spectral method clusters words of the same class, once they are sorted by their component in the eigenvector. Since we know that the spectral method detects the community structure, clusters are also likely to correspond to communities. With this aim, we define $S = \{1, \dots, i, \dots, n\}$ as a sequence of vertices where n is the number of vertices of the syntactic dependency network. We assume that S is the outcome of taking one eigenvector and sorting the vertices by the value of the corresponding components in the eigenvector. We define $C = \{c_1, \dots, c_i, \dots, c_n\}$, a boolean vector indicating, for every vertex in S , whether it can be of the class under consideration or it cannot. More precisely, we have $c_i = 1$ if the i -th vertex of S can be of class i and $c_i = 0$ otherwise. The correlations we are interested in are not computed between values in $X = \{x_1, \dots, x_i, \dots, x_n\}$ as in [14] but between values in C .

III. METHODS

For the spectral analysis, we will consider two different matrices, i.e. A and AA^T . We use AA^T because it is a way of obtaining equivalent vertices. Here equivalence means playing the same role in the network, i.e. pointing approximately to the same set of nodes. With this matrix, we are following the same approach as in [16]. A is used for two reasons: simplicity and as null hypothesis for AA^T . The methods in [15, 17] do not use directly A but close variations such as the Laplacian or the Normal matrix. The main difference between the present application of the spectral technique and the one performed in [14] are the following. First, we use simple adjacency matrices instead of matrices with edge weights. Second, we do not normalize the matrices. Normalizing the matrix is problematic here because most of verbs have no outgoing link (recall the convention adopted here is that arcs go from modifiers to heads). This would introduce special vertices in the graph without outgoing links. This would affect the statistics of the whole system, and this is why we did not normalize our adjacency matrix. Besides, nodes without outgoing links would mean trivial stationary solutions for a random walk on the network. This means that a walker moving on the graph would systematically get stuck in these “sink” vertices. Another solution pursued in [14] is to remove vertices with no in-

going or out-going links (pruning). Nevertheless, here pruning would mean a drastic reduction of verbs which may preclude the analysis of that class.

We introduce now various measures of the degree of clustering of words of a given class in C . We consider a special case of C , $C_{scrambled}$, sharing the same composition of C but with scrambled components. The measures obtained for $C_{scrambled}$ will be used as a null-hypothesis for those obtained for C . Significant clustering cannot be claimed unless the clustering obtained with C and $C_{scrambled}$ differ clearly. Scrambled sequences have been used to test the significance of long-range correlations in DNA sequences [37]. We used the first non-trivial 98 eigenvectors, so that we have 98 C vectors for each class.

We define k as a random variable measuring the length of a cluster of 1s in C . A cluster is a maximal sequence of consecutive elements equal to one in C . If $C = \{0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 1, 0, 1, 1, 1\}$, we have two clusters of length 3 (‘111’) and one cluster of length 1 (‘1’). We define $\langle k \rangle$ as the mean value of k in C .

We define $P(k)$ as the probability that a cluster has size k . We will estimate $P(k)$ from the proportion of clusters of length k in C . k is expected to follow a geometric distribution in $C_{scrambled}$. There, we have

$$P(k) = \langle c_i \rangle^k (1 - \langle c_i \rangle), \quad (2)$$

where $\langle c_i \rangle$ is the mean value of the components of C . In other words, $\langle c_i \rangle$ is the proportion of ones of C . From equation 2, it follows that

$$\langle k \rangle = 1/(1 - \langle c_i \rangle) \quad (3)$$

and the standard deviation is

$$\sigma(k) = \langle c_i \rangle^{1/2} / (1 - \langle c_i \rangle). \quad (4)$$

Since the estimated $P(k)$ may contain a considerable amount of noise, we will use, $P_{\geq}(k)$, the cumulative $P(k)$, defined as

$$P_{\geq}(k) = \sum_{K \geq k}^{\infty} P(K). \quad (5)$$

We are also interested in measures of the correlations between c_i and c_{i+d} , with $d > 0$. If the community detection clusters words of the same community consecutively in C [14], correlations above the expected value in the scrambled sequence are expected. Here we will use two measures of correlation: $\Gamma(d)$ and $I(d)$, that are, respectively, the Pearson correlation coefficient and the information transfer, between vertices at distance d in C [37]. $\Gamma(d)$ and $I(d)$ are complementary measures. $\Gamma(d)$ measures positive and negative correlations, whereas $I(d)$ cannot distinguish positive from negative correlations. $I(d)$ captures non-linear correlations that $\Gamma(d)$ does not detect [38]. $I(d)$ is apparently more sensitive to finite sampling than $\Gamma(d)$. Here, $\Gamma(d)$ is defined as

$$\Gamma(d) = \frac{COV(c_i, c_{i+d})}{\sigma(c_i)\sigma(c_{i+d})}, \quad (6)$$

where $1 \leq d < n$, $COV(c_i, c_{i+d})$ is the covariance between c_i and c_{i+d} , $\sigma(c_{i+d})$ is the standard deviation of c_{i+d} and $\sigma(c_i) = \sigma(c_{i+d})$ with $d = 0$. We have

$$\langle c_{i+d} \rangle = \frac{1}{n-d} \sum_{i=1}^{n-d} c_i. \quad (7)$$

The covariance is defined as

$$COV(c_i, c_{i+d}) = \sum_{i=1}^{n-d} \frac{(c_i - \langle c_i \rangle)(c_{i+d} - \langle c_{i+d} \rangle)}{n-d}. \quad (8)$$

The standard deviation is defined as

$$\sigma(c_{i+d}) = (\langle c_{i+d}^2 \rangle - \langle c_{i+d} \rangle^2)^{1/2} \quad (9)$$

as usual. Given that $\langle c_i^2 \rangle = \langle c_i \rangle$ because C is a binary sequence, it follows

$$\sigma(c_{i+d}) = (\langle c_{i+d} \rangle (1 - \langle c_{i+d} \rangle))^{1/2}. \quad (10)$$

Replacing equations 8 and 10 in equation 6, we obtain

$$\Gamma(d) = \sum_{i=1}^{n-d} \frac{(c_i - \langle c_i \rangle)(c_{i+d} - \langle c_{i+d} \rangle)}{(n-d)((\langle c_i \rangle (1 - \langle c_i \rangle))(\langle c_{i+d} \rangle (1 - \langle c_{i+d} \rangle)))^{1/2}}. \quad (11)$$

Here, $I(d)$ is defined as

$$I(d) = \sum_{x,y \in \{0,1\}} p(c_i = x, c_{i+d} = y) \log \frac{p(c_i = x, c_{i+d} = y)}{p(c_i = x)p(c_{i+d} = y)}, \quad (12)$$

where

$$p(c_i = x, c_{i+d} = y) = \frac{1}{n-d} \sum_{i=1}^{n-d} 1, \quad (13)$$

$c_i = x \text{ and } c_j = y$

$$p(c_i = x) = \sum_{y \in \{0,1\}} p(c_i = x, c_{i+d} = y) \quad (14)$$

and

$$p(c_{i+d} = y) = \sum_{x \in \{0,1\}} p(c_i = x, c_{i+d} = y). \quad (15)$$

Now we study the Romanian syntactic dependency network described in [25] with $n = 5563$ vertices. We choose that network from the set of three networks studied in [25] because it is the most complete. The other networks systematically lack several syntactic dependencies. The network is a small-world one with significantly high clustering, has a power-law distribution of vertex degrees and exhibits disassortative mixing. Those properties are shared with other non-linguistic biological networks [25]. As in [25], we worked on the largest connected component. A given word belongs to a given class if that word

| Type | Number | Proportion |
|------------|--------|------------|
| Verbs | 985 | 0.17 |
| Nouns | 3093 | 0.55 |
| Adverbs | 171 | 0.03 |
| Adjectives | 1129 | 0.20 |
| Other | 337 | 0.06 |

TABLE I: Counts of the frequency of very word class. The total amount of different words (i.e. vertices in the syntactic dependency network) is 5563. A word counts for a certain class if it has appeared at least once for that class in the Romanian corpus studied in [25]. Since a word can be of different classes, the sum of the column 'Number' does not necessarily equals the number of different words.

| Class | Cluster size | A | AA^T | Scrambled |
|------------|---------------------|------|-------------------|------------------|
| Verbs | $\langle k \rangle$ | 2.01 | 2.08 ± 0.32 | 1.21 ± 0.51 |
| | max | 43 | 24.96 ± 20.87 | 4.72 ± 0.84 |
| Nouns | $\langle k \rangle$ | 2.66 | 2.68 ± 0.13 | 2.25 ± 1.67 |
| | max | 32 | 34.35 ± 10 | 13.73 ± 2.04 |
| Adverbs | $\langle k \rangle$ | 1.06 | 1.08 | 1.03 ± 0.18 |
| | max | 2 | 3 | 2.16 ± 0.39 |
| Adjectives | $\langle k \rangle$ | 1.39 | 1.53 | 1.25 ± 0.56 |
| | max | 10 | 8.14 ± 0.83 | 5.11 ± 0.80 |

TABLE II: Cluster sizes for the four classes of words, the two types of matrices considered (A and AA^T) and scrambled vertex orderings. Standard errors smaller than 10^{-2} are not shown.

is labeled under that class at least once in the corpus that originated the syntactic dependency network in [25]. Table I shows the number of words in each class according the previous definition. Only 6% of words cannot fall in any of the previous classes. Participles and infinitives were excluded from the class verb, as it was done in the original corpus. Those words represent a very small fraction of the vertices.

IV. RESULTS

Fig. 2 shows some examples of C for nouns, adverbs and adjectives. Large clusters can be visually identified. Table II gives mean $\langle k \rangle$ and the maximum value of the cluster size over the sample set of eigenvectors.

Fig. 3 shows the mean $P(k)$ over the sample set of eigenvectors for the two kinds of matrices and the null hypothesis. For verbs and nouns, $P(k)$ obeys

$$P(k) \sim k^{-\gamma k} \quad (16)$$

for sufficiently large k . The agreement with equation 16 is lower adopting A rather than AA^T . Verbs and nouns cannot be given account of by the geometric distribution predicted by the null hypothesis, supporting thus the significance of the results. No conclusion can be made on adverbs, since they are not represented enough and it seems unlikely that adjectives follow equation 16.

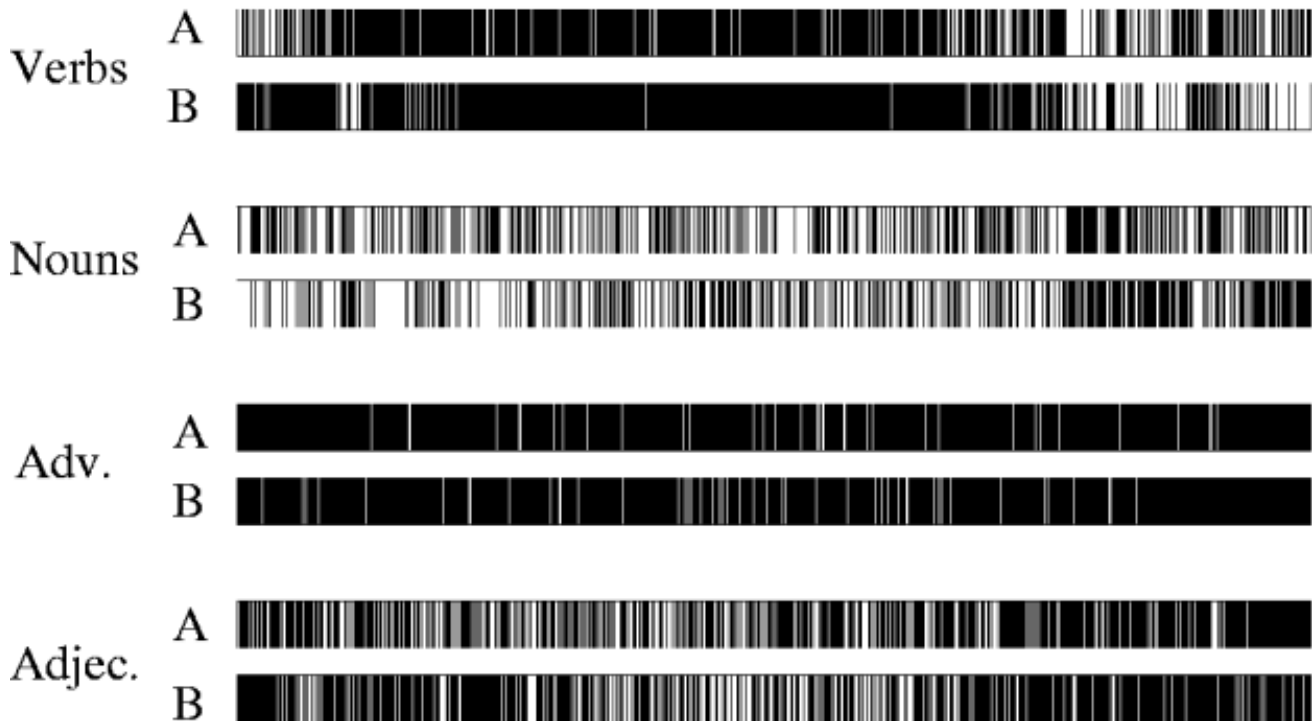


FIG. 2: Visual representation of the some binary vectors $C = \{c_i\}$ delivered by the spectral methods for different word classes: verbs, nouns, adverbs and adjectives. $c_i = 1$ if the i -th vertex (or word) belongs to the class under consideration and $c_i = 0$, otherwise. C has 5563 components, which is the number of vertices of the SDN. C is the outcome of ordering vertices by the the corresponding values of the eigenvectors delivered by the spectral methods and replacing the vertex by 1 if it is of the class under consideration or by 0 otherwise. White and black indicate, respectively, $c_i = 1$ and $c_i = 0$. Two types of C are shown for each word class: (A) an ordering using A and (B) ordering using AA^T .

Figure 4 and 5 show $\Gamma(d)$ for A and AA^T , respectively. Fig. 6 and 7 show $I(d)$ for A and AA^T , respectively. In order to determine the length of the correlations for a certain correlation measure in a conservative way, we define two series: $B_L(d)$ and $B_U(d)$. $B_L(d)$ is the mean value of a real correlation for distance d over the sample of eigenvectors minus the corresponding standard deviation. $B_L(d)$ is an approximate lower bound for the real value of the correlation. $B_U(d)$ is the mean value of the null hypothesis series over the sample of eigenvectors times the corresponding standard deviation. $B_U(d)$ is an approximate upper bound for the null hypothesis. Since real correlations tend to decrease with length (Figs. 4, 5 and 6 and 7), an approximate conservative measure of the length of statistically significant correlations is d^* , the smallest value of d at which $B_L(d)$ and $B_U(d)$ cross. Long-distance correlations were found (Table III), except for adverbs, due to the small amount of vertices that can be adverbs (recall Table I).

V. DISCUSSION

In the previous sections we have obtained three main results. First, the spectral methods clusters significantly

| Class | Correlation | d^* | |
|------------|-------------|-------|--------|
| | | A | AA^T |
| Verbs | $\Gamma(d)$ | 1164 | 915 |
| | $I(d)$ | 872 | 683 |
| Nouns | $\Gamma(d)$ | 50 | 63 |
| | $I(d)$ | 37 | 13 |
| Adverbs | $\Gamma(d)$ | 6 | 6 |
| | $I(d)$ | 1 | 1 |
| Adjectives | $\Gamma(d)$ | 148 | 881 |
| | $I(d)$ | 36 | 648 |

TABLE III: d^* , the approximate length of statistically significant correlations using two different correlation measures: the Pearson correlation coefficient $\Gamma(d)$ at distance d and the information transfer at distance $I(d)$. Four classes of words are considered.

words of the same class. Second, those spectral methods sort vertices in a way such that long-range correlations appear in the binary vector of membership to a certain class. Third, AA^T clusters word classes better than A . The chances of getting a large cluster are higher using AA^T (Fig. 3). The results presented here confirm the power of the spectral methods introduced in [14] for detecting community structure in linguistic networks. Us-

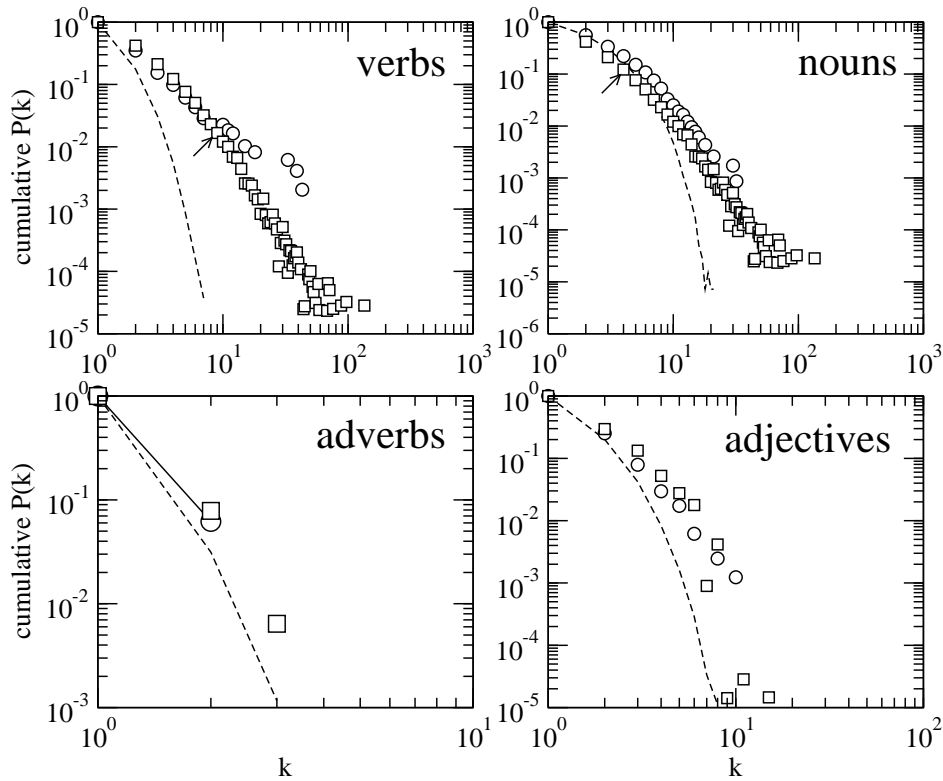


FIG. 3: Cumulative $P(k)$ where $P(k)$ is the proportion of clusters of length k . Results using A (circles), AA^T (squares) and the null hypothesis (dashed line) are shown. A clear power trend is found for the series of AA^T for verbs and nouns from the each arrow to the right.

ing a priori information about class membership of every vertex, we have discovered that the spectral method clusters words consistently with the word types that linguists have been distinguishing from a long time ago.

Discovering word classes minimizing the *a priori* amount of linguistic knowledge is a challenge in linguistics. Recently, Montemurro & Zanette [39] have clustered words using only information about the degree of heterogeneity with which words are distributed throughout a text. In particular, they have found that nouns and adjectives tend to be more heterogeneously distributed than verbs and adverbs. In a syntactic network, we have found that verbs and nouns and significantly more heteroge-

neously distributed according to the ordering provided by spectral methods than adverbs and adjectives. The present work suggests that word classes could be eventually discovered using only the structure of syntactic interactions.

Acknowledgments

We thank Ricard Solé for the drawing in Fig. 1. This work was supported by the FET Open Project COSIN (IST-2001-33555) and Integrated Project DELIS.

-
- [1] A.-L. Barabási and R. Albert, *Rev. Mod. Phys.* **74**, 47 (2002).
 - [2] M. Buchanan, *Small World: Uncovering Nature's Hidden Network* (Weidenfeld and Nicolson, London, 2002).
 - [3] G. Caldarelli, R. Marchetti, and L. Pietronero, *Europhysics Letters* **52**, 386 (2000).
 - [4] R. Pastor-Satorras and A. Vespignani, *Evolution and Structure of Internet: A Statistical Physics Approach* (Cambridge University Press, 2004).
 - [5] L. Adamic and B. Huberman, *Nature* **401**, 131 (1999).
 - [6] L. Adamic and B. Huberman, *Science* **287**, 2115 (2000).
 - [7] R. Albert, H. Jeong, and A. L. Barabási, *Nature* **401**, 130 (1999).
 - [8] A. Vazquez, A. Flammini, A. Maritan, and A. Vespignani, *Nature Biotechnology* **21**, 697 (2003).
 - [9] D. J. Watts, *Small-worlds: the dynamics of networks between order and randomness*, Princeton studies in complexity (Princeton University Press, Princeton, N.J., 1999).
 - [10] D. J. Watts and S. H. Strogatz, *Nature* **393**, 440 (1998).
 - [11] G. Caldarelli, A. Capocci, P. De Los Rios, and M. Muñoz, *Physical Review Letters* **89**, 258702 (2002).

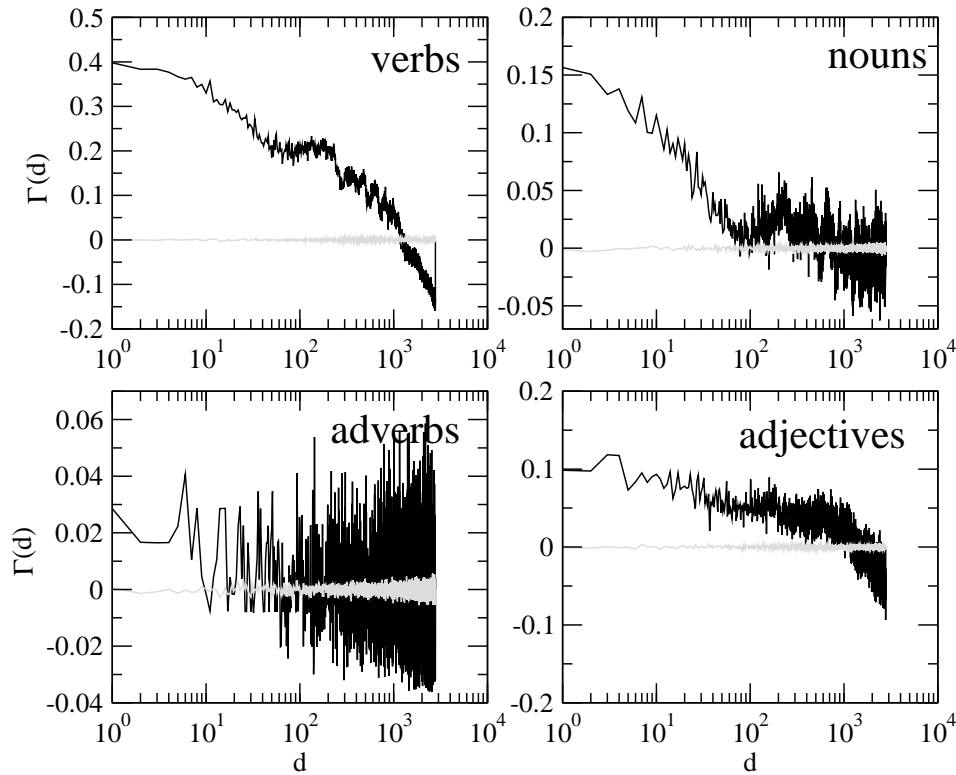


FIG. 4: $\Gamma(d)$, the correlation coefficient between words of a particular class as a function of d , the distance in a vertex ordering provided by the spectral methods. Two series are shown for each word class: $\Gamma(d)$ using A (black) and the $\Gamma(d)$ for the scrambled ordering (gray).

- [12] M. E. J. Newman, SIAM Review pp. 167–256 (2003).
- [13] S. N. Dorogovtsev and J. F. F. Mendes, Adv. Phys. **51**, 1079 (2002).
- [14] A. Capocci, V. D. P. Servedio, G. Caldarelli, and F. Colaiori, Physica A (2004), in press.
- [15] A. Seary and W. Richards, Proceedings of the International Conference on Social Networks **1: Methodology**, 47 (1995).
- [16] J. Kleinberg, Journal of the ACM **46**, 604 (1999).
- [17] K. M. Hall, Management Science **17**, 219 (1970).
- [18] F. Wu and B. A. Huberman, Eur. Phys. J. B **38**, 331 (2004).
- [19] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi, Proc. Natl. Acad. Sci. USA **101**, 2658 (2004).
- [20] M. E. J. Newman, cond-mat/0309508 (2003).
- [21] J. Reichardt and S. Bornholdt, cond-mat/0402349 (2004).
- [22] S. Fortunato, V. Latora, and M. Marchiori, cond-mat/0402522 (2004).
- [23] L. da Fontoura Costa, cont-mat/0405022 (2004).
- [24] L. Donetti and M. A. Muñoz, cond-mat/0404652 (2004).
- [25] R. Ferrer i Cancho, R. V. Solé, and R. Köhler, Physical Review E **69**, 051915 (2004).
- [26] M. Steyvers and J. Tenenbaum, cond-mat/0110012 (2001).
- [27] A. E. Motter, A. P. S. de Moura, Y.-C. Lai, and P. Dasgupta, Phys. Rev. E **65**, 065102 (R) (2002).
- [28] M. Sigman and G. A. Cecchi, Proc. Natl. Acad. Sci. USA **99**, 1742 (2002).
- [29] R. Ferrer i Cancho and R. V. Solé, Proc. R. Soc. Lond. B **268**, 2261 (2001).
- [30] I. Melčuk, *Dependency Syntax: Theory and Practice* (SUNY, Albany, 1988).
- [31] R. Hudson, *Word Grammar* (Blackwell, Oxford, 1984).
- [32] D. Sleator and D. Temperley, Tech. Rep., Carnegie Mellon University (1991).
- [33] I. Melčuk, in *International Encyclopedia of the Social and Behavioral Sciences*, edited by N. J. Smelser and P. B. Baltes (Pergamon, Oxford, 2002), pp. 8336–8344.
- [34] R. Ferrer i Cancho, J. Quantitative Linguistics (2004), submitted.
- [35] R. Ferrer i Cancho, Physical Review E **70** (2004).
- [36] M. E. J. Newman, Phys. Rev. E **67** p. 026126 (2003).
- [37] W. Li and K. Kaneko, Europhysics Letters **17**, 655 (1992).
- [38] W. Li, Journal of Statistical Physics **60**, 823 (1990).
- [39] M. A. Montemurro and D. Zanette, Advances in Complex Systems **5**, 7 (2002).

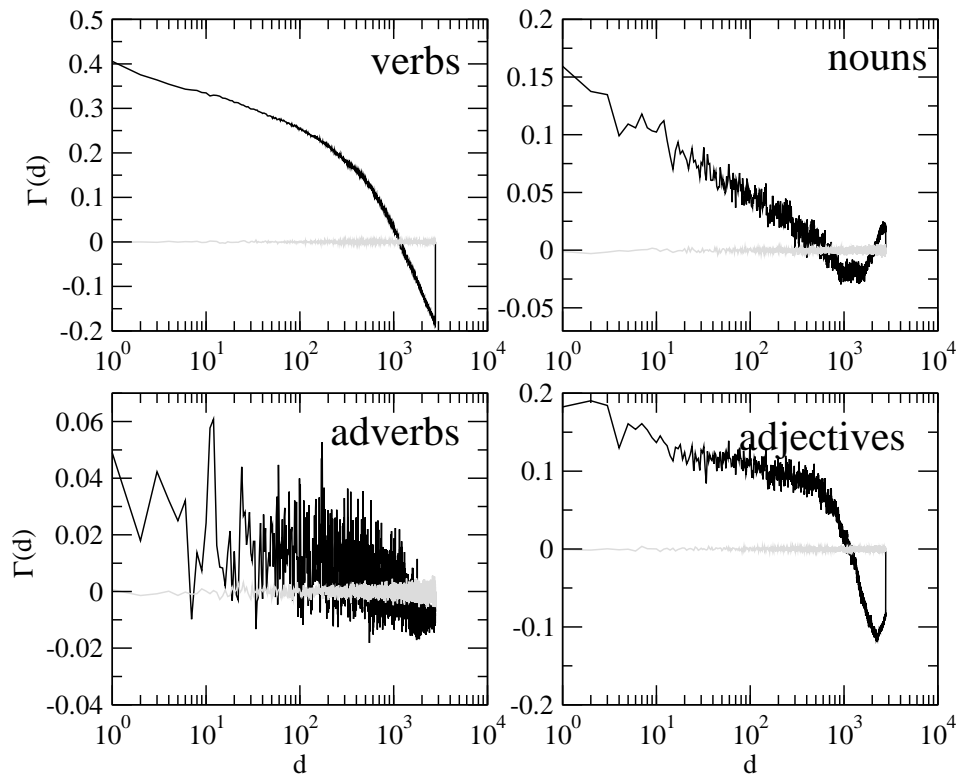


FIG. 5: $\Gamma(d)$, the correlation coefficient between words of a particular class as a function of d , the distance in a vertex ordering provided by the spectral methods. Two series are shown for each word class: $\Gamma(d)$ using AA^T (black) and the $\Gamma(d)$ for the scrambled ordering (gray).

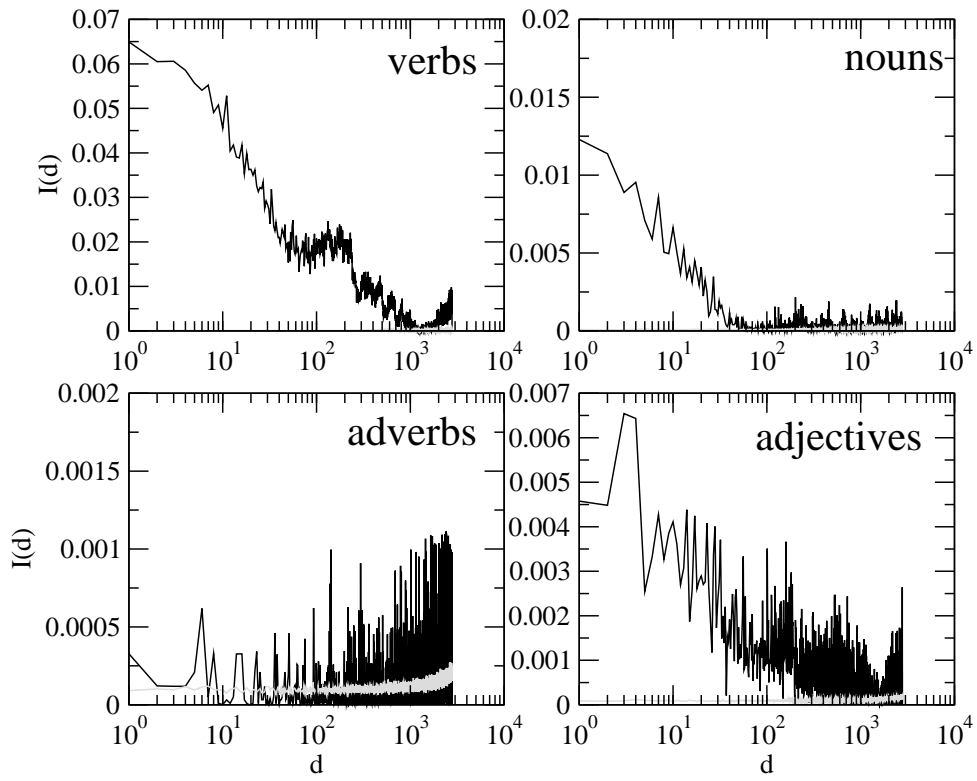


FIG. 6: $I(d)$, the information between words of a particular class as a function of d , the distance in a vertex ordering provided by the spectral methods. Two series are shown for each word class: $I(d)$ using A (black) and $I(d)$ for the scrambled ordering (gray).

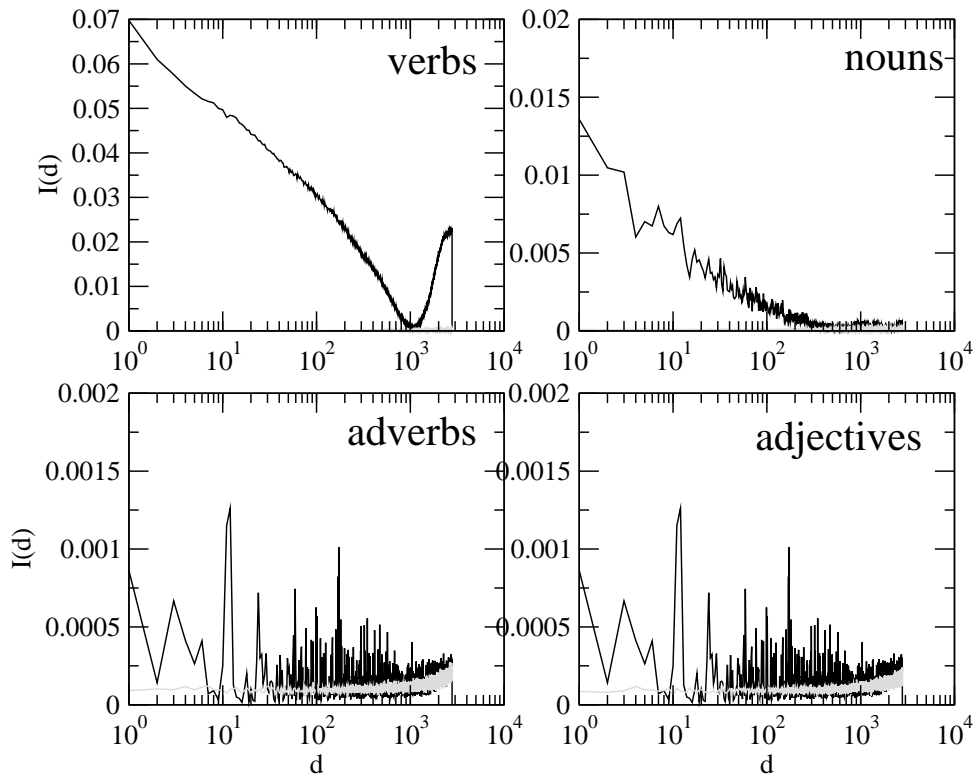


FIG. 7: $I(d)$, the information between words of a particular class as a function of d , the distance in a vertex ordering provided by the spectral methods. Two series are shown for each word class: $I(d)$ using AA^T (black) and $I(d)$ for the scrambled ordering (gray).