View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Universidad Carlos III de Madrid e-Archivo

PHYSICAL REVIEW E **84**, 046108 (2011)

Disentangling categorical relationships through a graph of co-occurrences

Juan Martinez-Romo,[1,*] Lourdes Araujo,[1,†] Javier Borge-Holthoefer,[2,‡] Alex Arenas,[2,3,§] José A. Capitán,[3,4,‖] and José A. Cuesta[4,¶]

[1]*Departamento de Lenguajes y Sistemas Informáticos, Natural Language Processing and Information Retrieval Group, Universidad Nacional de Educación a Distancia, 28040 Madrid, Spain*

[2]*Instituto de Biocomputación y Física de Sistemas Complejos, Universidad de Zaragoza, Mariano Esquillor, 50018 Zaragoza, Spain*

[3]*Departament d'Enginyeria Informàtica i Matemàtiques, Universitat Rovira i Virgili, 43007 Tarragona, Spain*

[4]*Grupo Interdisciplinar de Sistemas Complejos, Departamento de Matemáticas, Universidad Carlos III de Madrid, 28911 Leganés, Spain*

(Received 19 July 2011; revised manuscript received 15 September 2011; published 19 October 2011)

The mesoscopic structure of complex networks has proven a powerful level of description to understand the linchpins of the system represented by the network. Nevertheless, the mapping of a series of relationships between elements, in terms of a graph, is sometimes not straightforward. Given that all the information we would extract using complex network tools depend on this initial graph, it is mandatory to preprocess the data to build it on in the most accurate manner. Here we propose a procedure to build a network, attending only to statistically significan relations between constituents. We use a paradigmatic example of word associations to show the development of our approach. Analyzing the modular structure of the obtained network we are able to disentangle categorical relations, disambiguating words with success that is comparable to the best algorithms designed to the same end.

DOI: 10.1103/PhysRevE.84.046108 PACS number(s): 05.65.+b, 89.75.Fb, 89.75.Hc

I. INTRODUCTION

The recent burst and success of network modeling is not limited to the traditional niches of this framework. Nowadays, networks pervade almost all field of science, not only sociology and mathematics but also biology, physics, engineering, and neurosciences. This has been possible due to our current capabilities to collect and process large amounts of data, which in turn have evidenced that interactions in many natural, social, and manmade systems are accurately described by complex networks [1]. Examples of networks' transdisciplinary character include the spreading of diseases [2–4], robustness of gene regulatory networks [5], the emergence of cooperative behavior [6], and the diffusion of information in sociotechnical systems [7], to mention just a few.

In many cases the datasets are naturally arranged as a network. These cases are most amenable to be analyzed this way. For very important examples, though, the network structure is not evident. For instance, text semantics arises from relationships between words, but these relationships are not included in linguistic corpora. In these cases one of the most critical preprocessing steps is to unveil the hidden network between the elements of the dataset. In other cases the network which the data are arranged in is not necessarily the best one to extract the relevant information. Examples of this are the databases of customers and purchases from which one would like to extract information in order to make good recommendations. Recommending amounts to identifying profile among customers and building up a customer network based on the similarity between their profiles This is far from trivial [8]. As a matter of fact, creating a good recommender is still an open problem, as anyone purchasing in famous Internet shops knows.

The goal of this paper is to provide a method to construct a meaningful network in these cases. The abstract setup is define by some elements that are supposed to be related, but from which we only have indirect information. The kind of information we have can be described as the belonging of these elements to a given collection of sets. For instance, if our aim is to extract semantic connections between words, the information we have is whether or not words appear in each one of a given collection of texts (documents, paragraphs, sentences, etc.). Or, if we aim at devising a recommender system for an Internet store, our information is whether or not customers have bought each one of a catalog of products. Or, if we are investigating protein functionality, we should check whether two proteins participate or not in the same reaction —or metabolic or regulatory pathway.

This list is far from exhaustive, and the formalism we will provide has enough generality to account for these and many other examples. However, both for the sake of illustration and to evaluate the results, we will apply the formalism to just two examples: first to extract clusters of hashtags from Twitter messages which are related by meaning; second, to carry on semantic disambiguation of words in text. The firs example is just qualitative and is included here as an illustrative example. The second application has been chosen for two reasons: on the one hand, linguistic corpora are more easily accessible than, e.g., customer-product databases or other kind of data; on the other hand, researchers in natural language processing have devised benchmarks to evaluate the quality of many automatic task performers in this area—semantic disambiguation in our particular case—and this will allow us to perform a quantitative evaluation of the output. This notwithstanding, the procedure we provide should be regarded in its full generality. We hope these two examples will make this point clear.

We have organized this paper as follows: Sec. II describes the construction of a graph linking elements whose

*juaner@lsi.uned.es
†lurdes@lsi.uned.es
‡borge.holthoefer@gmail.com
§alexandre.arenas@urv.cat
‖jcapitan@math.uc3m.es
¶cuesta@math.uc3m.es

 ©2011 American Physical Society

co-occurrence in different sets is statistically significant which will be the basis of our method; Sec. III summarizes the philosophy of clustering in network communities, emphasizing a particular community detection algorithm that we will later apply; Sec. IV applies this formalism to the extraction of related hashtags from Twitter messages, as a qualitative test of the performance of this proposal; Sec. V is devoted to test the method proposed in a particular example, namely word sense induction (WSI), whose results in text disambiguation are evaluated with standard benchmarks and compared to alternative approaches; finall , Sec. VI summarizes the work.

II. CO-OCCURRENCE GRAPH

Suppose we have a set of elements $V = \{v_1, \ldots, v_n\}$ which will defin the nodes of a graph. These elements may or may not appear in each one of a given collection of sets $\mathfrak{S} = \{S_1, \ldots, S_N\}$. If these sets characterize relevant features of the elements v_i, a common way of definin an undirected weighted graph is to defin a binary vector $\mathbf{e}_i = (e_{i1}, \ldots, e_{iN})$ for each node v_i, where $e_{ij} = 1$ if the element v_i belongs to set S_j and $e_{ij} = 0$ otherwise, and then assign the weight $w_{ij} = \cos\theta_{ij} = \mathbf{e}_i \cdot \mathbf{e}_j/(\|\mathbf{e}_i\|\|\mathbf{e}_j\|)$ to the link from v_i to v_j. This technique has been widely used [9–13]. In the case we are going to consider here, the sets S_i do not characterize any feature but in a loose, statistical sense. This is the situation we face in the examples of the introduction. For instance, in word disambiguation, nodes are words and sets are documents, paragraphs, or sentences in a corpus. The appearance of a word in a document is usually related to its meaning (this is the idea we aim at exploiting) but not necessarily. We may easily fin in a document about, say, politics a sentence like "the early bird gets the worm," but this does not mean that "bird" or "worm" are meaningful political terms; their appearance in that document is just casual. Or, in the example of the recommendation, a customer of a bookshop may be very fond of science-fictio literature and yet buy a romantic poems book simply as a present for a friend. Obviously this adds no meaning to the profil of this particular customer; on the contrary, if this purchase is assigned too relevant a meaning it may distort future recommendations (most of us have suffered from this effect).

Thus, we need to assign a significanc to the co-occurrence of two elements in a certain number of sets out of the whole collection. This is akin to statistical hypothesis testing, the hypothesis being that the two words co-occur because of semantic relatedness. Statistical hypothesis testing relies on the setting of a null model that define what we consider pure chance. Ours will be one in which elements are randomly and independently distributed among the sets of the collection (see the Appendix). Co-occurrence will be considered statistically significan if it is unlikely that it arises by pure chance, i.e., is generated by the null model.

Given two elements (e.g., words) α and β that independently occur in n_α and n_β sets (e.g., documents), respectively, we want to test how likely it is that we would observe more than r sets with both elements α and β. Specificall , we want to calculate the probability (p-value of the null hypothesis)

$$p = \sum_{k \geqslant r} p(k) \tag{1}$$

that we would expect to observe more than r co-occurrences by chance, where $p(k)$ is given by Eq. (A4) [or in a more practical form by Eq. (A6)]. If $p \ll 1$ we can consider that the appearance of the two elements in the same set is statistically significant and therefore it is likely that this coincidence has some "meaning."

The way to proceed from here is standard in statistical hypothesis testing: a confidenc level p_0 is set (typically $p_0 \leqslant 0.05$, i.e., the null hypothesis is wrong with a reliability of 95% or larger) so that co-occurrence is meaningful only if $p < p_0$. Thus a link is define between elements i and j only if they co-occur according to this criterion. But the lower p the more significan the co-occurrence, so it makes sense to assign a strength to those links. A practical way of definin this strength is as $s = \log(p_0/p)$, which is tantamount to saying that $p = p_0 e^{-s}$. Strength is proportional to the order-of-magnitude difference between p and p_0. The graph constructed through this procedure will be referred to as the co-occurrence graph.

III. CLUSTERING IN COMMUNITIES

Once the co-occurrence graph is define we can proceed to cluster the nodes, for instance by performing a community decomposition. Communities are subgraphs such that nodes within modules exhibit some kind of structural or dynamic affinit between them, and therefore it is plausible to assume that every community shares a common meaning, different from that of the remaining communities.

A good deal of community detection algorithms are designed to optimize modularity [14–16], a magnitude that compares intracommunity connection density with that of randomized versions of the same graph. Recently, though, much attention is being devoted to diffusive algorithms [17–20]. The basic idea behind them is that a random walk gets trapped more easily in densely connected parts of the graph, which correspond to communities. The algorithm we will apply here is one of these, namely Pons and Latapy's *Walktrap* [17].

The rationale behind *Walktrap* is the following. If two nodes, i and j, belong to the same community, the probability that the random walker will visit j (i) starting from i (j) after t steps, P_{ij}^t (P_{ji}^t), must be high. Moreover, if these two nodes belong to the same module it is plausible to assume that $P_{il}^t \approx P_{jl}^t$, for any other vertex l; in other words, the accessibility of any node is somewhat similar from i and j. These facts amount to introducing a definitio of distance between two nodes as

$$d(i,j) = \sqrt{\sum_{l=1}^{n} \frac{\left(P_{il}^t - P_{jl}^t\right)^2}{k_l}}, \tag{2}$$

k_l denoting the degree (number of links) of vertex l (the degree in the denominator accounts for the fact that nodes with higher connectivity are more likely to be visited by the walker; see [17] for details). Thus, a small $d(i,j)$ indicates that nodes i and j belong to the same community. This distance can be define for any value of t. Computational costs as well as the exponential convergence speed of the random-walk process

suggest the choice of a small value t, which we have set to $t = 4$.

Pons and Latapy's proposal is particularly efficien because $d(i,j)$ can be easily generalized to a coarse-grained structure, where distance is measured not between vertices but between communities $d(C_1,C_2)$. Once the algorithm starts, vertices are merged into modules yielding an increasingly fast computation. Given that an exhaustive search over all possible partitions is unfeasible, a greedy heuristic is used such that at each time step the smallest variation of the distance $\Delta d(C_1,C_2)$ is chosen and the two communities merged into a larger one, in great resemblance to Newman's fast algorithm [15].

IV. FIRST APPLICATION: EXTRACTION OF RELATED HASHTAGS FROM TWITTER MESSAGES

We have obtained a collection of one million tweets from the Twitter social network. These tweets span a period of one month and cover topics about politicians, cars and motorbikes, rock bands, and banks. About 20% of these tweets contain a hashtag. Hashtags are words or phrases prefi ed with a hash symbol (#), with multiple words concatenated, such as those in "#Apple #Microsoft and oracle vs #google the reason is #android." They are meant to design specific previously introduced concepts, and therefore act as new coined words. Our tweets collection contains 13 500 different hashtags.

Because of its consisting of new terms it is impossible to extract semantic relations between hashtags from a corpus, as with natural language. Thus any procedure establishing relationships between them can be very useful regarding opinion mining [21], economic issues [22], or political activity [23,24]. Research is rapidly growing on this topic, and there are websites where the hashtag world can be explored (e.g., Ref. [25]).

We have constructed the co-occurrence graph for hashtags in our data by interpreting tweets as sets and hashtags as elements. Thus we have checked for co-occurrence of pairs of hashtags in the same tweet, and we proceed as explained in Sec. II for different thresholds p_0. Communities (clusters) are then detected in the resulting graph with the algorithm described in Sec. III.

Lacking a quantitative way of evaluating the result, we have selected a few hashtags with a clear meaning (e.g., #angelamerkel, #honda, #bmw, #Lego, #alcatel, etc.). Communities shrink their size and focus their "meaning" upon decreasing p_0. Table I and Fig. 1 illustrate the results for some of these hashtags (#angelamerkel, on one hand, and #bmw and #honda on the other) for two of the smallest values of p_0 that we have checked. Two facts are noticeable: first a clear semantic content emerges from these clusters (in the sense that all hashtags included in the cluster are more or less clearly related), and, second, decreasing p_0 significantl focus this semantic relatedness (for instance, in the case of #honda, nearly all hashtags in the right panel of Fig. 1 are connected to motorbikes, whereas #bmw is mainly connected to cars).

TABLE I. An example of the resulting clusters of hashtags obtained for two different significanc thresholds p_0. In boldface are the chosen hashtags whose cluster of hashtags related by meaning is looked for. (Recall that a link connects two hashtags if their co-occurrence in a tweet is a statistical event with a p-value smaller than p_0.)

$p_0 = 5 \times 10^{-5}$		$p_0 = 3 \times 10^{-7}$	
nikkei	denmark	italian	china
sweden	finlan	defici	zapatero
communist	mcdonald	telegraph	italy
ceiling	downgrade	davidcameron	dejager
EEC	silvioberlusconi	silvioberlusconi	G7
davidcameron	**angelamerkel**	sarkozy	merkel
ecb	china	bbc	BBC
G7	debt	berlusconi	debt
default	cac40	cac40	
merkel	ftse100	dax	
telegraph	BBC	ftse100	
dax	dejager	ecb	
sarkozy	zapatero	default	
wapo	italian	downgrade	
florenc	reuters	wapo	
berlusconi	bbc	EEC	
italy	defici	**angelamerkel**	

V. SECOND APPLICATION: WORD SENSE INDUCTION

The second test for the validity of the co-occurrence graph to infer "meaning" out of a dataset has been applying it to extract the different senses of ambiguous words from a set of texts in which they appear. The reason to choose this application—beyond its importance in the context of natural language algorithms—is that it allows for a quantitative evaluation of the results obtained, as well as a comparison with other algorithms performing the same task available in the specialized literature.

In what follows we will make a precise description of the problem as well as an evaluation contest that will be used as a benchmark. Then we will provide the details of the construction of the co-occurrence graph out of the corpus and of how to apply it to word sense extraction, and we will compare the results of the benchmark to alternative approaches that competed in the contest. In particular we will illustrate how the results improve—to a certain extent—as the threshold of statistical significanc is made more stringent.

A. Description

Word sense disambiguation (WSD) is a fundamental task in natural language processing. It amounts to identifying the particular meaning or sense of any polysemic word in a sentence or text. The quality of many other processes such as machine translation, question answering, or web search, depends on solving the WSD problem with a reasonable degree of accuracy. The most frequent approach to WSD relies on the context, i.e., the surrounding words, of the word to be disambiguated to choose the appropriate word sense. Depending on how words senses are defined WSD methods are classifie into supervised and unsupervised. Supervised WSD amounts to manually creating sources of information. These resources range from text annotated with word senses to machine-readable dictionaries. Probably the most popular of these resources is WordNet [26]. This is an electronic

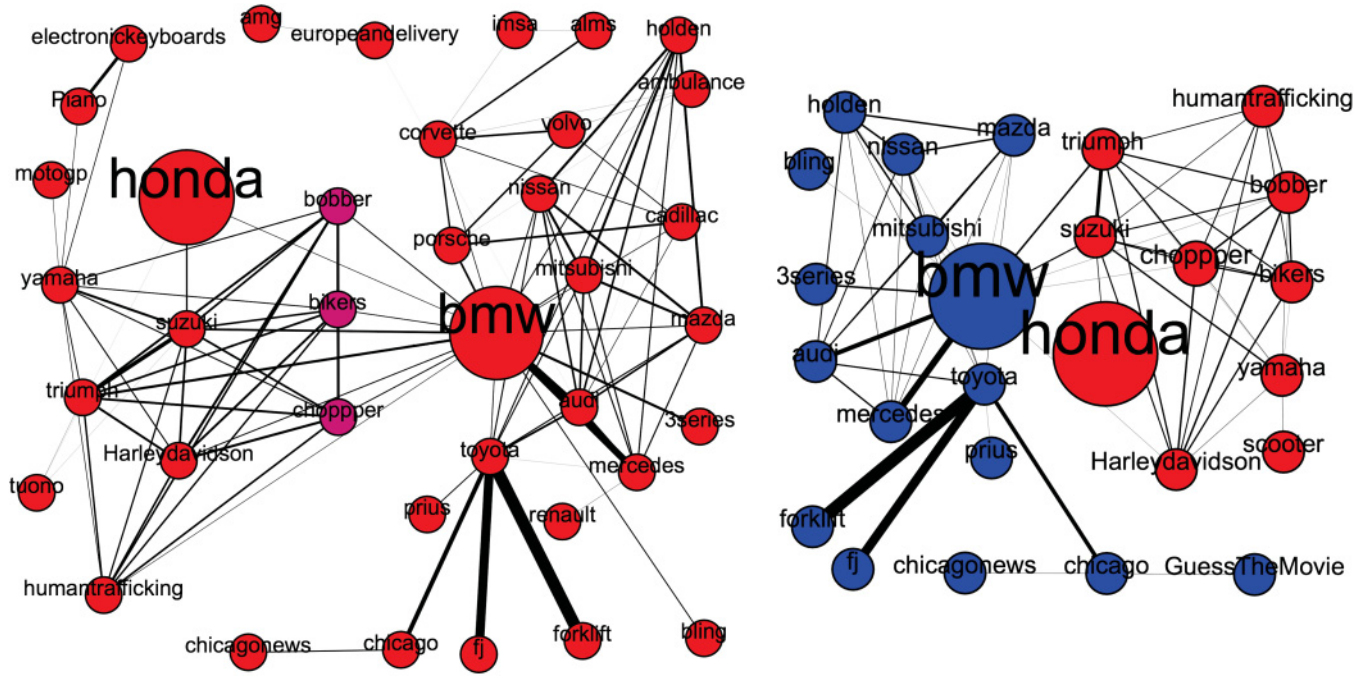

FIG. 1. (Color online) The figur illustrates the refinemen level obtained for two different significanc thresholds p_0, focusing on two specifi brands referenced by their corresponding hashtags: Honda, which is mainly a motorbike manufacturer, and BMW, which is mainly a car manufacturer. Left: If p_0 is permissive ($p_0 = 5 \times 10^{-5}$), the community detection algorithm assigns a single cluster for different vehicle manufacturers. However, it cannot distinguish car or motorbike constructors, because some generic hashtags (like #bikers or #chopper, with a slightly different color) and brands which build both cars and motorbikes act as mergers. Right: a lower significanc threshold ($p_0 = 3 \times 10^{-7}$) correctly splits the previous community in two.

dictionary of nouns, verbs, adjectives, and adverbs which organizes related concepts into synonym sets. These groups of synonyms represent a concept. Many WSD proposals have used WordNet as an inventory of the possible senses for a word. However, the manual creation of such resources is a very expensive and time-consuming task, and they are frequently unavailable when considering new languages or domains. Moreover, it is unlikely that a predefine sense inventory could be useful for different situations, since the nature and degree of sense distinction vary a lot with the applications [27].

On the contrary, unsupervised methods do not have any such predefine sense inventory; instead they perform what is known as word sense induction (WSI). Most works devoted to WSI are based on the so-called vector space model [10,28], which represents each instance of the considered word as a vector of features. These features are usually words appearing in the same context. Then vectors are clustered and the resulting clusters represent the word senses. Recently some works have developed graph-based methods to achieve WSI [13,29,30]. Typically these works select contexts of a given ambiguous word w and assign every word appearing in these contexts to a node of the graph. Then a link is established between two nodes if their words coincide in one or more contexts of w. Once the graph for the word w has been constructed, senses are also obtained by applying different graph clustering algorithms.

Our proposal lies in this latter category as it also define a co-occurrence graph, but it differs in at least two relevant aspects: first it ignores whether words are ambiguous or not and creates a single co-occurrence graph out of the whole corpus; second, and more importantly, co-occurrence is considered only if it is statistically significant and this significanc is reflecte in the link strength. In what follows we will evaluate our proposal and compare it to previous approaches for a standard benchmark in WSI.

B. Evaluation contest

SemEval (semantic evaluation) is an ongoing series of evaluations of computational semantic analysis systems [31] SemEval-2007 included an evaluation of WSI. Participants in the task were provided a dataset for the evaluation consisting of a set of annotated English texts taken from the *Wall Street Journal* and the *Brown Corpus*. These texts were hand-annotated with *OntoNotes* senses [32] instead of *WordNet*. The reason for that is that *OntoNotes* senses are coarser than those in *WordNet*, thus reducing the number of different senses to be induced. Participants were provided 100 target words (of which 65 were verbs and 35 nouns) and for each of them a set of contexts in which the target word appears. The officia corpus had 17 649 occurrences for the target nouns, and 12 200 occurrences for the target verbs. Participants were then given a set of examples and were asked to tag all of them with senses induced from the corpus. Examples contained at least one occurrence for every target word.

There are two alternatives [31] to decide whether a WSI system is better than another one. The firs one, named unsupervised evaluation by the contest organizers, compares the induced clusters and the clusters corresponding to "gold standard" senses extracted from a hand-annotated corpus produced by human judges with a good level of interjudge

agreement. In this case the system performance is quantifie by *F-Score,* the harmonic mean of *precision* and *recall.* Precision is define as the fraction of words that are correctly assigned to a given cluster, and recall is the fraction of words that are correctly assigned to the corresponding gold standard sense. Thus, according to F-Score, a perfect clustering solution would be one where there were a one-to-one mapping between clusters and gold standard senses. The second alternative, named supervised evaluation, maps the induced senses to gold standard senses and uses the mapping to tag the test corpus with gold standard tags. In this case, the system performance is given by the recall measure. (For further details about the precise evaluation procedure see [31].)

C. Co-occurrence graph of the SemEval-2007 corpus

Our aim is to achieve the WSI required by the SemEval-2007 contest by constructing a co-occurrence graph (as define in Sec. II) from the corpus provided by the organizers and later clustering it in communities.

Our goal is to create a graph out of the words of the corpus by linking every two words sharing a common meaning. Documents are coherent pieces of information, so that it is natural to assume that words appearing in the same document have high chances to have related meanings. We know, however, that this is not strictly true, so we can only be confiden that two words truly share a common meaning if they are *often* found in the same documents. Therefore our definitio of the co-occurrence graph (Sec. II) applies to this example.

As the most meaningful words are (common or proper) nouns and verbs, we extract from each document these words. This requires tagging a word in the corpus with its part-of-speech tag. To accomplish this we have employed the GENIA tagger [33,34]. A stemming process is then applied to the extracted words, reducing them to their stem with the aim of increasing the significanc of the number of occurrences. This is done by applying Porter's algorithm [35]. After these usual preprocessing steps, the algorithm considers each pair of words (stems) and computes the corresponding p-value from the number of documents in which each word appears and the number in which they appear simultaneously [c.f. Eq. (A6)]. If this value is below p_0, a link is created in the graph for the pair of words and a weight assigned according to the method described in Sec. II.

Walktrap [17], the algorithm to detect communities described in Sec. III, is then used to cluster words with related meaning. These communities represent the different senses of the target words. Each detected community is treated as a different sense of the corpus. We now need to assign one of these senses to each instance of a target word. We have tried several ways to measure similarity between an instance and a community and found that the one that provides the best results is the overlap between the instance text and the communities (i.e., the number of words appearing simultaneously in the instance and the community). If two or more communities get the same highest score, we select the one appearing more frequently in the other instances. Accordingly, the frequencies of every community are extracted at the beginning of the process and updated after each new assignment.

Communities have very different sizes. Some of them contain just a few terms and thus have a high level of specificity others have hundreds of words and are far too generic. Intuitively, words usually have just a few senses so one can expect not to have a large variability in the number of induced senses. A raw application of the above described scoring procedure renders a large number of senses, some of them having been assigned to just a few instances. In order to reduce this proliferation of induced senses we have carried out a postprocessing step filterin out every community whose sense has been assigned to less than 5% of the instances. These instances are then reassigned to the community with the next highest score.

The communities selected by applying this procedure determine the senses of the target words. Each instance is assigned to a unique sense, and thus all instances assigned to the same sense are part of the same cluster of results. The results can then be evaluated according to the standard SemEval-2007 procedure. Figure 2 sketches the detailed procedure described in this subsection.

D. Results

We present here the results of the above-described procedure for word sense induction, comparing them with those obtained by other systems that actually participated in the contest. This amounts to using the same datasets and the same measures of evaluation. According to the state of the art in WSI [31], unsupervised evaluation (F-Score) favors systems with a low number of senses, while supervised evaluation

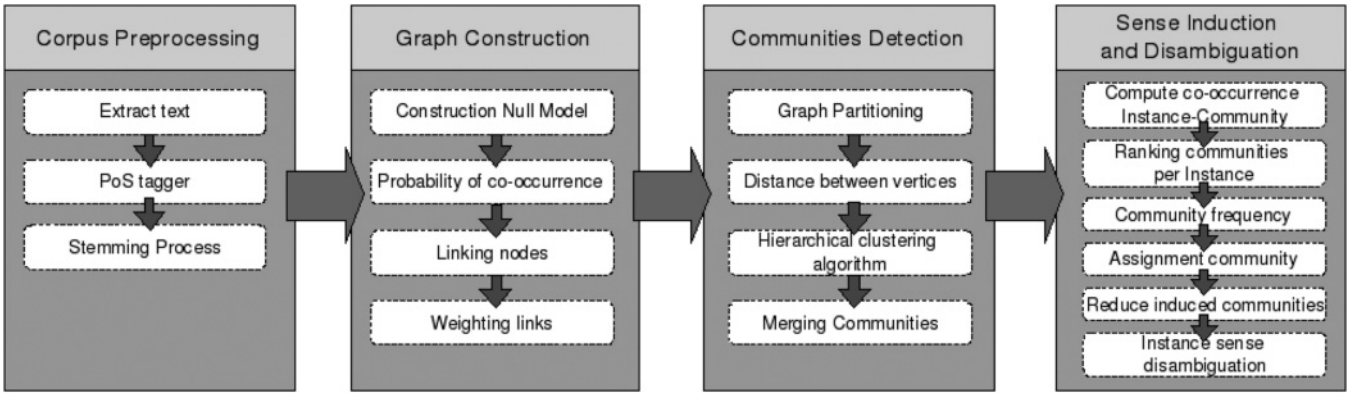

FIG. 2. Sketch of the co-occurrence graph-based approach for inducing and disambiguating word senses. It contains four main modules: (i) corpus preprocessing, (ii) graph construction, (iii) communities detection, and (iv) sense induction and disambiguation.

TABLE II. Unsupervised evaluation on the test corpus (F-Score) for all words (nouns and verbs). The "System" column corresponds to the names the participants of SemEval-2007 gave to their respective proposals. UBCAS was submitted by the task organizers. Acronyms denote the different competing proposals: UBCAS (University of Basque Country-Agirre-Soroa), COG (Co-Occurrence Graph, our proposal), UPVSI (Universidad Politécnica de Valencia-Sistemas de Información), UMND2 (University of Minnesota, Duluth), I2R (Institute for Infocomm Research), and UOY (University of York).

System	F-Score	Ranking
UBCAS	78.7	1
COG	**72.2**	**2**
UPVSI	66.3	3
UMND2	66.1	4
I2R	63.9	5
UOY	56.1	6

(supervised recall) favors systems with a high number of clusters. This is also verifie in Tables II and III: systems highly ranked in Table II got a low ranking in Table III and vice versa.

In the design of our system we have attempted neither to generate a fi ed number of senses nor to use information from the gold standard to fin tune the mean size of senses. The only tuning made to our system has been removing those communities that had a marginal number of instances assigned (the postprocessing step). Thus, the number of senses induced varies substantially depending on the target word.

An input parameter of our system is the significanc threshold p_0. The smaller this value the more stringent is the decision that co-occurrence of two words in the corpus is meaningful. Figure 3 shows the results reached for the two described measures (unsupervised F-score and supervised recall) using different values of p_0 spanning nine orders of magnitude. Two things are worth noticing: firs of all, supervised recall is rather insensitive to the value of p_0; secondly, unsupervised F-score significantl grows as p_0 decreases from 10^{-3} down to around 10^{-7} and then decreases. The latter result is remarkable because it stresses the fact that selecting only statistically significan relationships (with a very restrictive criterion indeed) clearly improves the performance of the algorithm, i.e., its ability to extract meaning. Beyond a certain significanc threshold, though, we may start losing relevant links, thereby degrading the performance.

Table II shows the unsupervised evaluation of the systems on the test corpus for all words (nouns and verbs). Our system is

TABLE III. Supervised evaluation on the test corpus (supervised recall) for all words (nouns and verbs). UBCAS was submitted by the task organizers. (See Table II for the meaning of acronyms.)

System	Supervised recall	Ranking
I2R	81.6	1
UMND2	80.6	2
COG	**79.9**	**3**
UPVSI	79.1	4
UBCAS	78.5	5
UOY	77.7	6

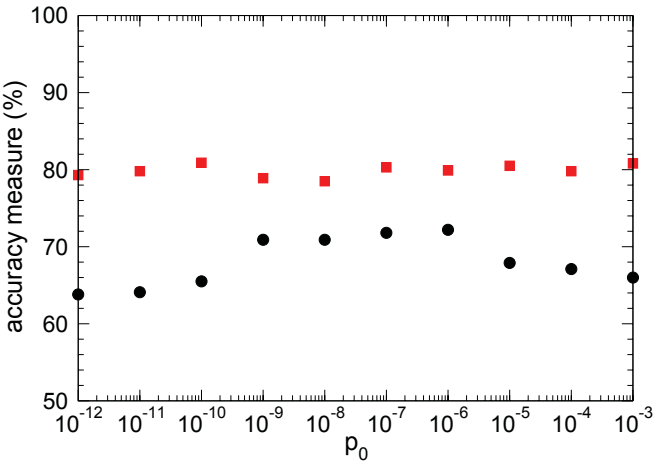

FIG. 3. (Color online) Accuracy measures reached by our method (circles: unsupervised F-Score; squares: supervised recall) as a function of the significanc threshold p_0.

also included and ranked in the second place. The significanc threshold value chosen has been $p_0 = 10^{-6}$, a value for which its performance is optimal according to Fig. 3. The F-Score obtained by our system is very highly ranked—especially so if we take into account that the system ranked in firs position was submitted by the contest organizers.

The results of the supervised evaluation can be seen in Table III. The evaluation is also performed over the test corpus for all words (nouns and verbs). The results of our system (denoted COG, an acronym for co-occurrence graph) are shown along with those of the participants. COG is ranked in third place.

A few remarks are worth noticing regarding these results. First of all, rankings in both measures, F-score and recall, are anticorrelated (if a system ranks high according to one measure, it ranks low according to the other). In particular, UBCAS, the system submitted by the contest organizers—the firs in F-score—is ranked fift according to recall. Likewise, I2R is ranked firs according to recall and fift according to F-Score. Second, differences in recall are much smaller than in F-Score. This is compatible with what is observed in Fig. 3. Recall measures all fluctuat (plus or minus 2%) around 80%. And third, the difference between COG and the next system ranked according to F-score is of around 6%. This is as much as the difference between the best system, UBCAS, and COG, but notice that no knowledge other than that provided by the corpus has been used to optimize our system—precisely because we aim at testing the idea of building a statistically significan co-occurrence graph to extract meaning.

Finally, Table IV shows the average number of senses for each target word of the gold standard and that found by

TABLE IV. Average number of senses for each target word. COG is an acronym for co-occurrence graph used to denote our proposal.

System	All	Nouns	Verbs
Gold standard	3.79	4.46	3.12
COG	2.89	2.97	2.85

COG. Note that COG generates less senses per target word than the gold standard. Taking into account that supervised evaluation favors the systems with a high number of clusters, it is significan that even with a low number of clusters COG performs so well. This is a hint that this procedure is a reliable method for truly identifying word senses.

VI. CONCLUSIONS

In this paper we have introduced a way to extract meaning out of a set of related data by exploiting the widely available algorithms for community detection on graphs. The procedure is based on the construction of a co-occurrence graph in which nodes are the elements of the dataset which one needs to extract meaning from, and links reflec co-occurrence in the collection of sets that form the database. While this method has already been used before, we introduce the important novelty of deciding on the existence of the link and assigning a weight according to the statistical significanc of this co-occurrence, taking as a reference that of a null model in which co-occurrence arises from pure chance. The method proposed here has important potential applications, such as word sense induction of texts or recommendation algorithms.

We have tested our proposal in two applications. In the firs one, a graph of the hashtags found in a collection of tweets of the Twitter social network has been constructed and used to infer relationships between those hashtags. The results, albeit only of a qualitative nature, very clearly illustrate both the validity of the procedure and the importance of the threshold for statistical significanc that define the graph. In the second application, the data of SemEval-2007, a contest organized to test systems of word sense induction of texts, have been analyzed with our algorithm. In contrast to the previous application, this one provides a quantitative way of evaluating the performance of the system and comparing it with state-of-the-art approaches to solve the same problem. Our results indicate, first that the significanc constraint imposed to accept a link does improve the quality of the network—hence of the results—and, second, that the method is able to reach a high overall performance without taking advantage (as other special purpose systems do) of the details of the task to be solved. These results make us confiden that the method can be successfully applied to other problems like designing recommendation algorithms, a fiel where much research is still needed to produce satisfactory results.

ACKNOWLEDGMENTS

We are very grateful to Damiano Spina Valenti for sharing with us his collection of tweets prior to publication. We acknowledge financia support through Grant No. FIS2009-13364-C02-01, Holopedia (Grant No. TIN2010-21128-C02-01), MOSAICO (Grant No. FIS2006-01485), PRODIEVO (Grant No. FIS2011-22449), and Complexity-NET RESINEE, all of them from Ministerio de Educación y Ciencia in Spain, as well as support from Research Networks MODELICO-CM (Grant No. S2009/ESP-1691) and MA2VICMR (Grant No. S2009/TIC-1542) from Comunidad de Madrid, and Network 2009-SGR-838 from Generalitat de Catalunya.

APPENDIX: NULL MODEL

Let N denote the number of sets in the collection and n_1 and n_2 the number of them in which the firs and second elements are found, respectively. The null model amounts to selecting these sets randomly and independently among the N total sets and obtaining the probability distribution of k, the number of sets in which both elements coincide. Selecting firs n_1 and then n_2 sets out of N, forming the collection can be done in

$$\binom{N}{n_1}\binom{N}{n_2} \tag{A1}$$

different ways. Let us denote by k the number of coincidences between the firs and second selection of sets. This number must be in the range

$$\max\{0,n_1+n_2-N\} \leqslant k \leqslant \min\{n_1,n_2\}. \tag{A2}$$

This expresses the fact that there can be zero coincidences only if the sum n_1+n_2 does not exceed N—otherwise there will be at least n_1+n_2-N coincidences—and that the largest number of coincidences cannot exceed the smallest number of selections, n_1 or n_2.

Let us now count in how many of these choices there are exactly k coincidences. We can classify sets into four kinds: k sets showing a coincidence, n_1-k sets selected only in the firs choice, n_2-k sets selected only in the second choice, and $N-n_1-n_2+k$ sets—provided this number is nonzero—not selected in any of the two choices. Thus the sought number will be given by the multinomial coefficien

$$\binom{N}{k,n_1-k,n_2-k} \tag{A3}$$

[we use the definitio $\binom{p}{q_1,\ldots,q_n} \equiv p!/q_1!\cdots q_n!(p-q_1-\cdots-q_n)!$]. Accordingly, the probability that exactly k sets coincide when we chose firs n_1 and then n_2 sets randomly and independently among the N total sets will be

$$p(k)=\binom{N}{n_1}^{-1}\binom{N}{n_2}^{-1}\binom{N}{k,n_1-k,n_2-k} \tag{A4}$$

if $\max\{0,n_1+n_2-N\} \leqslant k \leqslant \min\{n_1,n_2\}$ and $p(k)=0$ otherwise.

Equation (A4) is not computationally practical, but we can write it down in a more convenient form. For that purpose we introduce the notation $(a)_b \equiv a(a-1)\cdots(a-b+1)$, for any $a \geqslant b$, and without loss of generality we assume that $n_1 \geqslant n_2 \geqslant k$. Then

$$\begin{aligned} p(k) &= \frac{(n_1)_k(n_2)_k(N-n_1)_{n_2-k}}{(N)_{n_2}(k)_k} \\ &= \frac{(n_1)_k(n_2)_k(N-n_1)_{n_2-k}}{(N)_{n_2-k}(N-n_2+k)_k(k)_k}, \end{aligned} \tag{A5}$$

where in the second form we have used the identity $(a)_b = (a)_c(a-c)_{b-c}$ valid for $a \geqslant b \geqslant c$. Equation (A5) is better written as

$$p(k)=\prod_{j=0}^{n_2-k-1}\left(1-\frac{n_1}{N-j}\right)\prod_{j=0}^{k-1}\frac{(n_1-j)(n_2-j)}{(N-n_2+k-j)(k-j)}. \tag{A6}$$

[1] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D. Hwang, Phys. Rep. **424**, 175 (2006).

[2] R. Pastor-Satorras and A. Vespignani, Phys. Rev. Lett. **86**, 3200 (2001).

[3] V. Colizza, R. Pastor-Satorras, and A. Vespignani, Nat. Phys. **3**, 276 (2007)

[4] S. Gómez, A. Arenas, J. Borge-Holthoefer, S. Meloni, and Y. Moreno, Europhys. Lett. **89**, 38009 (2010).

[5] F. Li, T. Long, Y. Lu, Q. Ouyang, and C. Tang, Proc. Natl. Acad. Sci. USA **101**, 4781 (2004).

[6] J. Gómez-Gardenes, M. Campillo, L. M. Floría, and Y. Moreno, Phys. Rev. Lett. **98**, 108103 (2007).

[7] J. Ratkiewicz, S. Fortunato, A. Flammini, F. Menczer, and A. Vespignani, Phys. Rev. Lett. **105**, 158701 (2010).

[8] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich, *Recommender Systems: An Introduction* (Cambridge University Press, Cambridge, 2010).

[9] K. McRae, G. S. Cree, M. S. Seidenberg, and C. Mcnorgan, Behav. Res. Meth. Ins. C. **37**, 547 (2005).

[10] H. Schütze, Comput. Linguist. **24**, 97 (1998).

[11] A. Purandare and T. Pedersen, in *Proceedings of the Conference on Natural Language Learning 2004*, edited by H. T. Ng and E. Riloff (Association for Computational Linguistics, Stroudsburg, 2004), pp. 41–48.

[12] J. Artiles, A. Peñas, and F. Verdejo, in *Proceedings of Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text, Senseval-3, ACL-SIGLEX*, edited by R. Mihalcea and P. Edmonds (Association for Computational Linguistics, Stroudsburg, 2004), pp. 58–63.

[13] I. P. Klapaftis and S. Manandhar, in *Proceedings of the 18th European Conference on Artificia Intelligence*, edited by M. Ghallab, C. D. Spyropoulos, N. Fakotakis, and N. Avouris (IOS, Patras, 2008), pp. 298–302.

[14] M. Girvan and M. E. J. Newman, Proc. Nat. Acad. Sci. USA **99**, 7821 (2002).

[15] M. E. J. Newman, Phys. Rev. E **69**, 066133 (2004).

[16] L. Danon, J. Duch, A. Arenas, and A. Díaz-Guilera, J. Stat. Mech. (2005) P09008.

[17] P. Pons and M. Latapy, Lect. Notes Comput. Sci. **3733**, 284 (2005).

[18] M. Rosvall and C. T. Bergstrom, Proc. Nat. Acad. Sci. USA **105**, 1118 (2008).

[19] J. Borge-Holthoefer and A. Arenas, Eur. Phys. J. B **74**, 265 (2010).

[20] Y. Kim, S. W. Son, and H. Jeong, Phy. Rev. E **81**, 016103 (2010).

[21] A. Pak and P. Paroubek, in Proceedings of LREC 2010 (unpublished).

[22] J. Bollen, H. Mao, and X. Zeng, J. Comp. Sci. **2**, 1 (2011).

[23] J. Borge-Holthoefer *et al.*, PLoS One **6**, e23883 (2011).

[24] M. Conover, J. Ratkiewicz, M. Francisco, B. Gonçalves, A. Flammini, and F. Menczer, in *Proceedings of the 5th International Conference on Weblogs and Social Media* (2011), pp. 81-96.

[25] [http://hashtagify.me].

[26] C. Fellbaum, ed., *WordNet: An Electronic Lexical Database* (MIT Press, Cambridge, 1998).

[27] T. Pedersen, in *Word Sense Disambiguation: Algorithms and Applications*, edited by E. Agirre and P. Edmonds (Springer, New York, 2006), pp. 133–166.

[28] T. Pedersen and R. F. Bruce, in *Proceedings of the Fifteenth National Conference on Artificia Intelligence*, edited by American Association for Artificia Intelligence (MIT Press, Cambridge, 1998), pp. 800–805.

[29] J. Véronis, Comput. Speech Lang. **18**, 223 (2004).

[30] E. Agirre, O. L. de Lacalle, D. Martinez, and A. Soroa, in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, edited by D. Jurafsky and E. Gaussier (Association for Computational Linguistics, Stroudsburg, 2006), pp. 585–593.

[31] E. Agirre and A. Soroa, in *SemEval '07: Proceedings of the 4th International Workshop on Semantic Evaluations*, edited by E. Agirre, L. Màrquez, and R. Wicentowski (Association for Computational Linguistics, Stroudsburg, 2007), pp. 7–12.

[32] E. Hovy, M. Marcus, M. Palmer, L. Ramshaw, and R. Weischedel, in *NAACL-Short '06: Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, edited by R. C. Moore, J. A. Bilmes, J. Chu-Carroll, and M. Sanderson (Association for Computational Linguistics, Stroudsburg, 2006), pp. 57–60.

[33] Y. Tsuruoka, Y. Tateishi, J.-D. Kim, T. Ohta, J. McNaught, S. Ananiadou, and J. Tsujii, in *Advances in Informatics*, edited by P. Bozanis and E. Houstis (Springer, Berlin, 2005), pp. 382–392.

[34] Y. Tsuruoka [http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/tagger/].

[35] M. F. Porter, Program **14**, 130 (1980).