Citraro and Rossetti *Applied Network Science* (2020) 5:55
https://doi.org/10.1007/s41109-020-00302-1

Applied Network Science

RESEARCH Open Access

Identifying and exploiting homogeneous communities in labeled networks

Salvatore Citraro[1,2*] and Giulio Rossetti[2]

*Correspondence:
salvatore.citraro@phd.unipi.it
[1]Department of Computer Science, University of Pisa, Pisa, Italy
[2]KDD-Lab, ISTI-CNR, Pisa, Italy

Abstract

Attribute-aware community discovery aims to find well-connected communities that are also homogeneous w.r.t. the labels carried by the nodes. In this work, we address such a challenging task presenting EVA, an algorithmic approach designed to maximize a quality function tailoring both structural and homophilic clustering criteria. We evaluate EVA on several real-world labeled networks carrying both nominal and ordinal information, and we compare our approach to other classic and attribute-aware algorithms. Our results suggest that EVA is the only method, among the compared ones, able to discover homogeneous clusters without considerably degrading partition modularity.

We also investigate two well-defined applicative scenarios to characterize better EVA: i) the clustering of a mental lexicon, i.e., a linguistic network modeling human semantic memory, and (ii) the node label prediction task, namely the problem of inferring the missing label of a node.

Keywords: Labeled community discovery, Network homophily, Node label prediction

Introduction

Node clustering, also known as community discovery, is one of the most important and productive tasks in complex network analysis. It is considered one of the most challenging subjects of the field, due to its intrinsic *ill-posedness*: the absence of a single, universal definition of *community* leads to a heterogeneous landscape of alternative approaches, each one redefining the task w.r.t. different topological criteria (density, bridge detection, feature propagation...). In such a complex scenario, a *meta*-definition of community discovery helps us to overcome the wide range of possible definitions:

Definition 1 (Community discovery). Given a network G, a community $c = \{v_1, v_2, \ldots, v_n\}$ is a set of distinct nodes of G. The community discovery problem aims to identify the set $\mathcal{C}$ of all the communities in G.

Community discovery algorithms aim to find well-connected, possibly well-separated, clusters of nodes. While classic approaches are driven only by the graph structure, nowadays, a large number of models includes extra features to complement the expressive

power of topology with other external information. *Feature-rich networks* (Interdonato et al. 2019) is the term proposed for a generalization of all these models, including temporal, probabilistic and attributed (or labeled) networks.

In the new context of feature-rich networks, classic approaches to community discovery are not enough. In this work, we are particularly interested in *labeled* or *node-attributed networks*, where reliable external information is added to the nodes as categorical or numerical attributes[1]. Node-attributed graphs are a quite useful model of social networks, and several salient homophilic dimensions (age, gender, nationality…) can be meaningfully identified and exploited by leveraging clustering approaches. Partitioning a social network by only considering social ties might produce well-defined and densely connected communities: classic approaches often assume an intrinsic homophilic behaviour between individuals, but they do not explicitly guarantee homogeneity while searching for meso-scale topologies. Thus, node attributes can improve the community discovery task by leveraging both topological and homophilic clustering criteria – an enhancement under the name of labeled community discovery:

Definition 2 (**Labeled community discovery**). Let $\mathcal{G} = (V, E, A)$ be a node-attributed graph where V is the set of vertices, E the set of edges, and A a set of nominal or ordinal attributes such that $A(v)$, with $v \in V$, identifies the set of labels associated to v. The labeled community discovery problem aims to find a node clustering $\mathcal{C} = \{c_1, ..., c_n\}$ of $\mathcal{G}$ that maximizes both topological clustering criteria and label homophily (or homogeneity) within each community.

Such a new task focuses on obtaining structurally well-defined partitions that also result in label-homogeneous communities. In this work, we address the labeled community discovery problem by introducing EVA (namely, LOUVAIN *Extended to Vertex Attributes*) (Citraro and Rossetti 2020). Our approach extends the well-known LOUVAIN algorithmic schema, designing a multi-criteria optimization approach aimed at exploiting both structure and semantic information while searching for node clusters.

The work is organized as follows. In "The EVA algorithm" section, we present the rationale of EVA. In "Experiments" section, we apply the method on real-world datasets: we show the possibilities of EVA to manage both nominal and ordinal attributes as well as its fruitful application in other domains than social systems, e.g., linguistic networks. In "EVA as a tool: node label prediction" section, we also show how the proposed algorithm can be applied, successfully, to address the task of node label prediction. In "Related work" section, we discuss the literature relevant to the work. Finally, "Conclusion" section concludes the paper and summarizes its main results as well as future research directions.

The EVA algorithm

The main goal of EVA is to identify homogeneous communities in complex node-attributed networks while assuring low modularity degradation w.r.t. standard topological modularity optimization-based approach. In the following, we will explain the algorithm rationale, we will study its complexity, and we will describe how it can handle both nominal and ordinal attributes.

[1] To prevent confusion between the words *attribute* and *label*, in the following we use *label* as a generic term for external information attached to a node, formally encoded as a *value* of an *attribute*.

Algorithm rationale. The algorithmic schema of EVA extends the LOUVAIN one (Blondel et al. 2008), a well-known bottom-up and hierarchical approach based on *modularity* optimization.

Definition 3 (Modularity)**.** Modularity measures the strength of the division of a network into sets of well-separated clusters or modules. It takes values in [-1/2, 1] and it is calculated as the sum of the differences between the fraction of edges that actually fall within a given community and the expected fraction if edges were randomly distributed. Formally:

$$Q = \frac{1}{(2m)} \sum_{vw} \left[A_{vw} - \frac{k_v k_w}{(2m)} \right] \delta\left(c_v, c_w\right) \tag{1}$$

where m is the number of edges, $A_{v,w}$ is the entry of the adjacency matrix for $v, w \in V$, k_v, k_w the degree of v, w and $\delta\left(c_v, c_w\right)$ is an indicator function taking value 1 iff v, w belong to the same community, 0 otherwise.

Modularity is one of the most used and discussed criteria among the methods based on an optimization function (Fortunato and Hric 2016). Its maximization is NP-complete, and it *suffers* of the well-known resolution limit (Fortunato and Barthelemy 2007), namely when it is worth considering two connected communities as a single one.

EVA, as LOUVAIN, uses modularity to update communities incrementally; moreover, it uses a quality function tailored to capture label homogeneity within communities, namely *purity*.

Definition 4 (Purity)**.** Given a community $c \in \mathcal{C}$ its purity is the product of the frequencies of the most frequent attributes carried by its nodes. Formally:

$$P_c = \prod_{a \in A} \frac{\max_{a \in A} \left(\sum_{v \in c} a(v) \right)}{|c|} \tag{2}$$

where A is the label set, $a \in A$ is a label, $a(v)$ identifies an indicator function that takes value 1 iff $a \in A(v)$. Purity takes values in [0,1] and it is maximized when all the nodes within a community share the same attribute value sequence. Finally, the purity of a partition is defined as the average of all the community purities:

$$P = \frac{1}{|C|} \sum_{c \in C} P_c \tag{3}$$

Purity assumes node labels as independent and identically distributed random variables; thus, considering the product of maximal frequency labels is equivalent to computing the probability that a randomly selected node in the given community has exactly that specific label profile.

EVA combines the two fitness function linearly, optimizing the following score:

$$Z = \alpha P + (1 - \alpha) Q \tag{4}$$

where α is a tuning parameter. It balances the importance of each component and adapts the algorithm results to the analyst needs.

EVA pseudocode is explained in Algorithm 1 and Algorithm 2. EVA takes as input a node-attributed graph $\mathcal{G}$ and the parameter α, and it returns a partition $\mathcal{C}$. As a first step

Algorithm 1 EVA

```
1: function EVA(G, α)
2:     C ← Initialize(G)
3:     Z ← αP + (1 − α)Q
4:     Z_prev ← −∞
5:     while Z > Z_prev do
6:         C ← MoveNodes(G, C, α)                    ▷ Ref. Algorithm 2
7:         G ← Aggregate(G, C)
8:         Z_prev ← Z
9:         Z ← αP + (1 − α)Q
10:    return C
```

Algorithm 2 EVA - MoveNodes

```
1: function MoveNodes(G, C, α)
2:     C_best ← C
3:     repeat
4:         for all v ∈ V(G) do
5:             s ← |P[v]|
6:             s_best ← size
7:             g_best = −∞
8:             for all u ∈ Γ(v) do
9:                 C_new ← C
10:                C_new[v] ← C[u]
11:                size_new ← |C_new[v]|
12:                q_gain ← Q_{C_new} − Q_C
13:                p_gain ← P_{C_new} − P_C
14:                g ← αp_gain + (1 − α)q_gain
15:                if g > g_best or g == g_best and s_new > s then
16:                    g_best ← g
17:                    s_best ← s_new
18:                    C_best ← C_new
19:    until C == C_best
20:    return C_best
```

(line 2, Algorithm 1) EVA assigns each node to a singleton community and computes the initial quality Z as a linear combination of modularity and purity. After the initialization step, the algorithm main-loop is executed (lines 5-9, Algorithm 1). As LOUVAIN, two phases compose EVA, i.e., (i) a greedy identification of a community, moving nodes such that they reach the optimal increase of quality, and (ii) the network agglomeration. EVA cycles over the graph *nodes* and evaluates the gain of quality while moving a single neighbouring *node* to its community (lines 8-14, Algorithm 2). For each pair (v, w) the local gain produced by the move is compared with the best gain identified so far and, if needed, the value is updated to keep track of the newly identified optimal quality: in case of ties, the move that results in a higher increase of the community size is preferred (lines 15-18, Algorithm 2). The procedure is repeated until no more moves are possible (line 19, Algorithm 2). As a result of Algorithm 1, the original allocation of nodes to communities is updated.

After this step, the aggregate function (line 7, Algorithm 1) hierarchically updates the original graph $\mathcal{G}$, transforming the communities in nodes, allowing to repeat the loop until no moves can optimize the quality score (lines 8-9, Algorithm 1).

Ordinal attributes support. We also extend the explained approach to cope with ordinal attributes, where order among node labels matters. For instance, the *education level*

of individuals in a social network can be modeled as an attribute with a set of n possible labels with an inner, natural order (e.g., high school degree < bachelor degree < master degree < PhD). In this scenario, the labeled community detection task imposes to find homogeneous clusters, grouping people according to both their social ties (topology) and to their education level (attribute). Considering such an attribute as a nominal may lead to a situation where an overall high educated cluster can indifferently contain minority nodes belonging to all the possible range of attribute values: EVA would insert these nodes in the cluster with the same probability. Thus, we can force an order among labels and *adjust* the distribution of minority nodes.

We model such an order as a linear distance. Formally:

Definition 5 (Linear distance)**.** Let O be an ordered chain of k labels, $O = \{l_0, l_1, \ldots, l_k\}$ and $l_i, l_j \in O$ then the linear distance among them is computed as:

$$d_{i,j} = 1 - \frac{|l_i - l_j|}{k} \tag{5}$$

Such a distance function accounts for the evaluation of nodes belonging to minority labels in terms of how distant the label is from the majority one.

As an example, given $O = \{l_a, l_b, l_c, l_d\}$, a cluster C having *purest* label l_a and two neighboring nodes i, j whose labels are, respectively l_b and l_d, iff the increase in modularity and size is the same, node i will be preferred for inclusion in C since $d_{l_a,l_b} < d_{l_a,l_d}$.

Such an approach allows to (semantically) minimize the degradation in terms of community purity when searching for a high-quality modularity partition. Indeed, the proposed distance function is applicable only if specific assumptions are satisfied (e.g., in the presence of a complete linear ordering among the values of an attribute). However, EVA support for ordinal attributes is designed to be parametric on the distance function used, thus making easy to define any label-specific distance (e.g., considering, for instance, hierarchical tree-like attribute values).

EVA complexity. EVA is a LOUVAIN extension. It shares the same performances in terms of time complexity, $O\left(|V|log|V|\right)$. Regarding space complexity, the difference w.r.t. LOUVAIN is related only to the data structures used for storing node labels. Considering k labels, the space required to associate them to each node in $\mathcal{G}$ is $O(k|V|)$: assuming $k << |V| < |E|$, we get a space complexity of $O(|E|)$.

Experiments

In this section, we firstly describe the experimental framework used to evaluate EVA performances, then we apply the proposed approach to several case studies, underlining its flexibility as an analytical tool.

Datasets

We applied EVA to several real-world datasets having different characteristics: here, we describe them, specifying the particular setup they provide for the application where they are proposed.

Single-nominal. Datasets in this category are characterized by a single node-attribute of nominal type, $|A| = 1$.

- *Cora* (McCallum et al. 2000) (2708 nodes, 5279 edges) is a co-citation network among computer science research papers, each of them labeled with one of 7 possible topics.
- *G+* (22355 nodes, 29032 edges) is a subgraph of the Google+ social network (Leskovec and Mcauley 2012), in which people are labeled according to their education degree (12 different levels).
- *IMDB* (Neville et al. 2003) (1169 nodes, 20317 edges) is a network of movies, each of them labeled with a binary value that identifies a node as a blockbuster or not.
- *Sense2vec* (Trask et al. 2015) (5309 nodes, 15170 edges) is a complex semantic network built on top of word vectors pre-trained on Reddit comments, where labels identify the part of speech of the words, i.e., a noun, a verb, an adjective, etc.
- *Web Spam* (Castillo et al. 2007) (9100 nodes, 505000 edges) is a network of linked web pages, indicating if they are spam or not.

Multi-nominal. Datasets in this category are characterized by multi node-attributes of nominal type, $|A| > 1$.

- *Amherst* (2235 nodes, 90954 edges) is one of the 100 social networks of *Facebook100* dataset (Traud et al. 2012); nodes have five different attributes, and we choose to mark from *Amh1* to *Amh5* the same network enriched from 1 to 5 node attributes. The attributes are *gender*, *status*, *dormitory*, *graduation year* and *major*: the order is maintained during the node profile enrichment, i.e., *Amh1*: *gender*, *Amh2*: *gender-status*, etc.

Ordinal. Nodes having a single attribute whose values are logically ordered characterize the graphs in this category – to prevent confusion, nine different attributes are considered, never in a multi-attribute scenario.

Mental lexicon (Stella et al. 2018): a collection of several layers of word similarities, in detail:

- a *semantic network* (3105[2] nodes, 26482 edges), namely a collection of three semantic layers, including synonyms, hierarchical relations and free associations;
- a *phonological network* (1683 nodes, 3731 edges), namely a layer for phonological similarities between word forms.

Each graph instance is considered w.r.t. one of the nine psycholinguistic dimensions from the *Glasgow Norms* (Scott et al. 2019), namely Arousal, Valence, Dominance, Concreteness, Imageability, Familiarity, Age of acquisition, Semantic size, and Gender association, plus the Length of the words. Each word is labeled (w.r.t. to the specific dimension) with a continuous scale measured by Likert scale (Likert 1932). To be able to perform the ordinal strategy of EVA, we discretized the labels, preserving their natural order.

Analytical results: single-nominal and multi-nominal

In this section, we will focus on the following questions:

1) does have the resolution limit (Fortunato and Barthelemy 2007) some effect on EVA, i.e.:

[2]It is a filtered graph composed by nodes having a corresponding label, see below.

  - does increasing α affect the number of communities identified?
  - does increasing α affect the size of the largest community?

2) does increasing α affect the topological quality of partitions in favor of purity?
3) does the multi-nominal scenario strongly affect EVA results?

The motivation behind 1) is that EVA does not use only modularity to optimize the quality of clusters; thus, the resolution limit effect can be *reduced* when the parameter is set in favour of purity. Also, in 2), we study α as a trade-off between connectivity and homogeneity; in 3), we want to focus explicitly on the multi-attribute profiles.

Cardinality of partition set and largest communities.

We tested EVA varying the values of α in the range [0,1] with bins of 0.1. We obtained 11 partitions $\mathcal{C}_\alpha$. The number of communities in $\mathcal{C}_\alpha$, namely the partition cardinality, is $|\mathcal{C}_\alpha|$. We recall that by definition (i) $\mathcal{C}_0$ is equivalent to the LOUVAIN partition and (ii) $\mathcal{C}_1$ is the set of the biggest connected components whose nodes share the same label profiling.

Figure 1a-b summarizes that (i) the number of communities of each partition normalized w.r.t. $|\mathcal{C}_0|$ underlines a significant fragmentation w.r.t. the increasing of α, and (ii) the size of the largest community normalized w.r.t. the size of that of $\mathcal{C}_0$ decreases w.r.t. the increasing of α until high values of the parameter (e.g., $\alpha = 0.8$) are reached, whereby the normalized size drastically increases. Such an increase depends on the distribution of labels overall the network: EVA, used with the highest values of α, tends to find quite perfectly homogeneous clusters without considering structure; thus, assortative zones might be merged even if they are not densely connected. Newman's assortativity coefficient values (w.r.t. the categorical attribute) (Newman 2003) are quite high for all the networks considered, i.e., $r = 0.76$ for *Cora*, $r = 0.34$ for *G+*, $r = 0.47$ for *IMDB* and $r = 0.59$ for *Sense2vec*. Even if Newman's assortativity is a global measure and it is not able to identify outliers and different mixing patterns in a graph, we can take these scores as an overall index of network homophily. Thus, high assortative scores justify the size of the largest community for high values of α.

Modularity vs. purity.

The results previously observed are coherent w.r.t. an incremental modularity decrease. Here, we discuss how a good clustering needs to achieve a reasonable trade-off between modularity and purity; thus, we study how the partitions that EVA generates vary w.r.t. α.

Figure 1c-e reports, in a two y-axes chart, modularity-purity values as functions of α. We observe that modularity lines tend to decrease in favour of a homogeneity function not based on the discovery of densely connected modules. Intuitively, a modularity line is stable if the value of the function for the partition $\mathcal{C}_\alpha$ is equal or quite close to that for $\mathcal{C}_{\alpha-0.1}$, while purity continues to increase. In other words, the stability of modularity indicates that communities are well-defined while EVA tries to make them homogeneous. Enforcing homogeneity contributes to producing less modular communities: recalling that EVA $\alpha = 1$ merges low-quality modules of a completely homogeneous subgraph, the analyst needs to pay attention to the modularity decrease.

As the figure shows, the partitions obtained with $\alpha = 0.7$ and $\alpha = 0.8$ still harmonize the two fitness functions in the *Sense2vec* dataset, because modularity degradation is not relevant while purity increases; a good trade-off is still reached with $\alpha = 0.6$ in the *G+* network, and with $\alpha = 0.7$, in *Cora*. The *IMDB* dataset was not shown in the absence of relevant degradation.

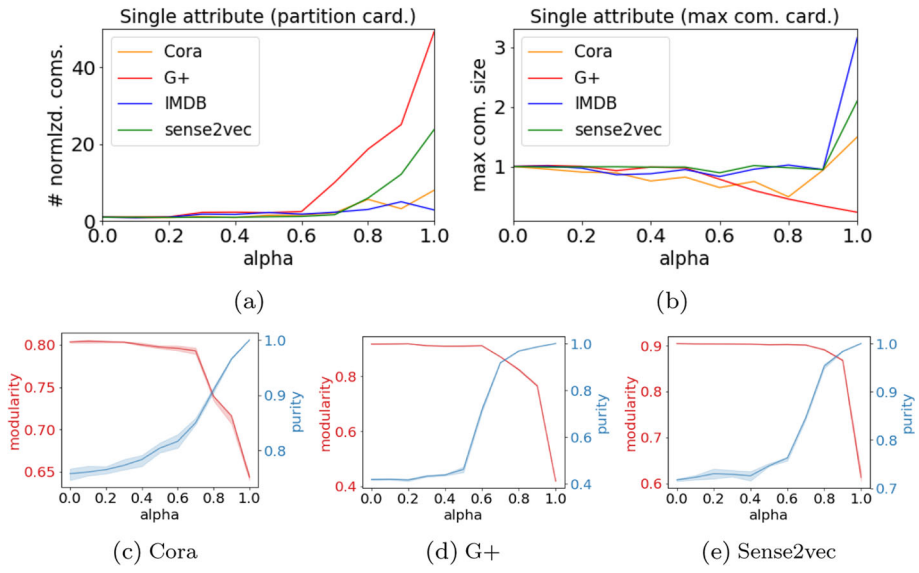


Fig. 1 **a** Number of communities and **b** size of the largest community as functions of α. **c**-**e** Modularity-Purity as a function of α, where the lines represent the average values over ten executions, and the surrounding areas are the interquartile ranges. EVA works well until the decrease of modularity values (i.e., the red trends): for instance, in G+, it starts from $\alpha = 0.6$, coherent with a relevant increase of partition cardinality

In Table 1, EVA is compared with four state-of-art community detection algorithms, among classic (LOUVAIN (Blondel et al. 2008), LEIDEN (Traag et al. 2019)) and attribute-aware approaches (SAC1 (Dang and Viennet 2012), STOC (Baroni et al. 2017)). There are several reasons to compare EVA to standard modularity-based methods as LOUVAIN and LEIDEN, for instance (i) a comparison of modularity decreasing w.r.t. the algorithms designed to maximize such a quality function and (ii) an evaluation of the implicit purity quantified by these only topological-based approaches. In detail, LOUVAIN is the natural baseline of EVA: it maximizes *only* modularity within the same algorithmic schema described for EVA in "The EVA algorithm" section. LEIDEN is designed to improve LOUVAIN, guaranteeing well-connected communities, whereas LOUVAIN fails to split bridges or disconnected components into separate groups. Moreover, we choose SAC1 and STOC for the (attribute-aware) comparison since they share similar methodologies. As EVA, they simultaneously consider structure and attributes during the clustering phase. In detail, SAC1 uses a composite function as a multi-objective criterion, i.e., a linear combination of modularity and an attribute similarity function (e.g., the Euclidean distance for numerical attributes or the matching coefficient for categorical ones). Differently, STOC defines a distance function as a linear combination of two distances, one for the topology and one for the attributes. The analyst has to define an *attraction ratio* α (decomposed in a topological attraction ratio α_T and a semantic attraction ratio α_S) trough which the algorithm is able to auto-tune the parameters involving the two distance functions.

In the table, we observe how LOUVAIN and LEIDEN can slightly guarantee higher modularity values than EVA, at the cost of a relevant gap of homogeneity. Thus, as long as we quantify just a slight modularity decrease in favour of purity in EVA, we can say that our method outperforms classic approaches. Moving to the attribute-aware methods, SAC1 obtains the lowest modularity partitions and never outperforms EVA in terms of purity.

Table 1 Single-nominal: comparison of modularity and purity scores. SAC1 was not able to terminate on the G+ in reasonable time due to its high computational complexity

	Modularity				Purity			
	Cora	G+	IMDB	Sense2vec	Cora	G+	IMDB	Sense2vec
LOUVAIN	.80	.91	.71	.90	.75	.42	.73	.71
LEIDEN	.80	.92	.71	.90	.75	.42	.72	.74
SAC1	.00	-	.00	.00	.49	-	.68	.79
STOC	.07	.45	.04	.00	.96	.66	.87	.52
$EVA_{0.5}$	.79	.91	.71	.90	.78	.45	.73	.72
$EVA_{0.8}$	.74	.82	.71	.89	.89	.90	.86	.94
$EVA_{0.9}$	.76	.76	.71	.86	.96	.98	.94	.98

Being an approach also based on modularity optimization, we did not expect these lower results. For the sake of clarity, we specify that SAC1 outputs were similar for all the values of its parameter, so we reported only $\alpha = 0.5$ in the table. STOC does not deal with modularity optimization, but it uses a distance function for capturing node similarity. The auto-tune of the parameters involving the distance equations depends on the values of α_T and α_S. Similarly to SAC1, several combinations of values gave similar results; thus, for the comparison, we used $\alpha_T = \alpha_S = \alpha = 0.5$, that is one of the solution proposed in the reference paper (Baroni et al. 2017). STOC is able to cluster the *G+* dataset, obtaining a reasonable trade-off between modularity and purity; however, not well-connected communities are discovered for *Cora*, *IMDB* and *Sense2vec*, looking at the modularity scores. A fair parenthesis should be open for STOC: being an approach explicitly designed to cluster large networks (as explained in the relative paper (Baroni et al. 2017)), good results on our small-medium networks can be hard to obtain.

Finally, to test the statistical significance of our results, we compared each partition against the ones we should expect by randomly clustering graph nodes. As already said, we noticed that the number of communities increases for high values of the parameter (see Fig. 1a). Since we also noticed this behaviour depends on the increased number of small-sized communities, we want to be sure that the purity scores are significant. In detail, we compared the purity of each partition (computed by Eq. 3) against the distribution of purity values of random permutations of the community labels. Thus, we performed a z-test, considering as a null model the distribution of purity of the random partitions: the more the partition purity deviates from the null model, the less the homogeneity within communities can be obtained by random permutations of the same number of labeled nodes. Formally:

$$z = \frac{P - \mu_p}{\sigma_p} \tag{6}$$

where P is the purity of the partition (the values for each network are those reported in Table 1), μ_p is the average purity score of the random partitions, and σ_p, the standard deviation.

The rationale underlining the proposed z-score comes from the Blockmodel Entropy Significance Test (BESTest) introduced in Peel et al. (2017), where the authors perform a statistical test to provide an estimate of how often a given partition obtains lower-entropy

Table 2 Single-nominal: statistical significance of the partitions

	z-score				*p*-value			
	Cora	G+	IMDB	Sense2vec	Cora	G+	IMDB	Sense2vec
LOUVAIN	50.53	22.47	14.99	22.16	0.00	0.00	0.00	0.00
LEIDEN	45.21	19.34	13.55	25.87	0.00	0.00	0.00	0.00
SAC1	1.04	-	1.56	3.76	0.14	-	0.05	0.00
STOC	20.20	57.17	3.02	0.00	0.00	0.00	0.001	0.5
EVA$_{0.5}$	61.59	37.88	15.62	22.06	0.00	0.00	0.00	0.00
EVA$_{0.8}$	45.98	55.49	10.83	18.51	0.00	0.00	0.00	0.00
EVA$_{0.9}$	39.80	69.04	7.77	15.41	0.00	0.00	0.00	0.00

explanation of the data, as viewed through a generative model[3]. Moreover, the BESTest is a generalization of the indicator Θ, proposed in Bianconi et al. (2009) to assess how much the topology of a network depends on a particular assignment of node characteristics.

Table 2 reports the z-scores of our proposed variant, with relative *p*-values, for all the datasets and algorithms. The purity of nearly all the partitions – of EVA, LOUVAIN and LEIDEN, in particular – is significant. STOC did not find any community in the *sense2vec* dataset; thus, the purity of the single giant community is exactly the same as that of the supposed random community/ies (i.e., z-score equal to 0.0). In this very extreme case, the purity of 0.52 (see Table 1, the cell STOC-*sense2vec*) corresponds to the most frequent value overall the network (for the sake of clarity, the *noun* part-of-speech). The not statistically significant *p*-values for SAC1 indicates that the purity values reported in Table 1 (e.g., 0.68 of *IMDB* or 0.49 of *Cora*) can also be obtained by the random permutations of the community labels.

Multi-nominal attribute profiles.

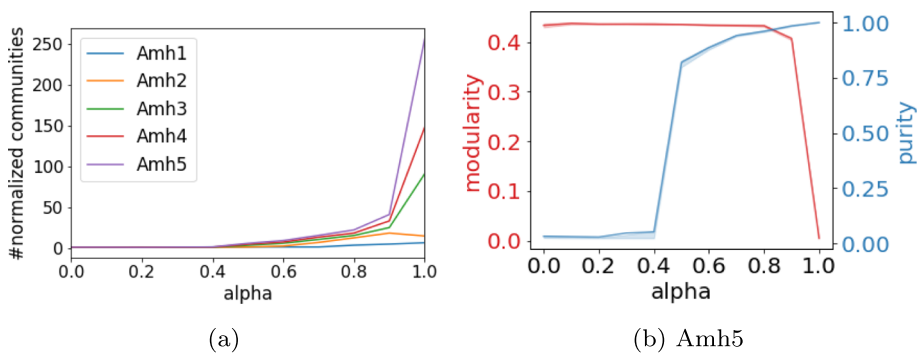
In a multi-nominal scenario, the nodes are enriched with more than one attribute. To address such an issue, EVA can exploit the general definition of purity as the product of the most frequent attribute values within a community. The questions we would like to answer are the following: does the number of distinct attributes affect the proposed approach in terms of (i) the number of communities discovered and (ii) modularity-purity trends?

Figure 2 confirms and extends what we already observed for the single attribute case: the number of communities increases with the varying of α and with the number of attributes considered. The number of attributes also affects modularity-purity trends, and modularity decreasing is similar to that observed for the single-attribute datasets. For space reasons, in Fig. 2 we report only the most complex scenario among the tested ones (i.e., *Amh5*, having $|A| = 5$); however, all the trends identified behave alike, as summarized in Table 3, where we show a comparison of modularity-purity lines for several instances of EVA on the *Amherst* network while varying the number of node attributes.

Empirical results: ordinal strategy

As previously introduced, homogeneity is strictly related to *homophily*, the tendency of nodes to interact with similar others by the dimensions defined by the attributes. Social

[3]For the sake of clarity, the versatility of the test proposed in Peel et al. (2017) would allow us to consider also non-generative community detection methods like those based on modularity optimization (see their Supplementary Materials, B.4)



(a) (b) Amh5

Fig. 2 Multi-nominal: **a** number of communities obtained w.r.t. different number of attributes and **b** modularity-purity trend in *Amherst* labeled with 5 attributes

homophily is the most well-known and well-studied example of network similarity, but it is not the only one possible. In this subsection, we provide an analysis of a particular type of complex system, under the name of linguistic networks. In such networks, nodes model words and links represent several types of linguistic relations based on the levels of the conventional linguistic analysis (e.g., phonology, syntax, semantics…).

As introduced in 2, we will focus on semantic and phonological networks, where homophily (i.e., external information encoded as attributes), models the tendency of words to connect with similar others in respect of different lexical and psychological properties. Homophily, here, defines the probability that a pair of connected nodes correlates with the presence of similar values of a linguistic variable. We observed similar behaviours studying the network of word vectors (i.e., *sense2vec* (Trask et al. 2015)), where connected clusters of word vectors were homogeneous w.r.t. the part of speech.

To the best of our knowledge, this is the first study in which a labeled community discovery approach is directly used to study a mental lexicon enriched with the set of psycholinguistic dimensions available from the *Glasgow Norms* (Scott et al. 2019). In doing so, we aim to observe whether well-connected semantic domains or phonological clusters are also homogeneous w.r.t. several psychological properties. To achieve such goal, we executed EVA on several instances of the same structure (as many as the attributes of the dataset) while using a different attribute one at a time. Indeed, we expect that polysemous words, namely words that convey more than one meaning, are clustered in different communities according to the specific attribute used.

As an example to explain our intuition and our framework of analysis, let us consider the word *star*: focusing on the *Semantic size*[4] variable, we should cluster the word with other ones related to cosmos and large objects in the universe; focusing on the *Arousal*[5] variable, we should cluster the word with other ones related to the metaphoric meaning of *brightness* and *shine*.

In a semantic network, the structure is not enough to group nodes. Such limitation is due to polysemy, namely the fact that a word can refer to different meanings according to the context in which it is used. Indeed, exploiting psychological and lexical variables of words (and using them as a complement to the structure), allows EVA to insert a

[4] I.e., a measure of something's dimensions, magnitude, or extent: high values refer to objects or concepts that are large, low values refer to objects or concepts that are small.
[5] I.e., a measure of excitement versus calmness: high values refer to words conveying excitement or that are stimulating and awaking, while low values refer to calming, relaxing or sleeping words.

Table 3 Multi-nominal: summary of modularity and purity scores on the *Amherst* network labeled with different number of attributes

	Modularity					Purity				
	Amh1	Amh2	Amh3	Amh4	Amh5	Amh1	Amh2	Amh3	Amh4	Amh5
$\text{EVA}_{0.1}$	.43	.43	.43	.43	.43	.49	.13	.09	.04	.03
$\text{EVA}_{0.5}$	.43	.43	.43	.43	.43	.49	.73	.77	.79	.80
$\text{EVA}_{0.8}$	.43	.42	.42	.42	.43	.95	.93	.95	.94	.96
$\text{EVA}_{0.9}$	.42	.36	.38	.40	.40	.97	.95	.97	.95	.98

polysemous word into the best cluster that is homogeneous w.r.t. the specific attribute considered.

To be able to perform EVA, we exploited the natural order among the *Glasgow Norms* variable values, measured according to a Likert scale (Likert 1932) (e.g., each word is characterized by an average rating given by a continuous value). We rounded such values to discretize them and be able to exploit the EVA ordinal strategy explained in "The EVA algorithm" section): we maintain the natural order among ratings, and we assume, for instance, there is more homogeneity between two words rated with two high values than two words rated with a high value and a low one, for all the scales.

A first example is shown in Table 4, where EVA (executed using several variables at a time) is compared to LOUVAIN. There, we focused on which words are in the same clusters of *star*, explaining why the only topological information is not able to focus on one single meaning of a polysemous word.

In Fig. 3a we observe, according to the ordinal strategy, that in the majority of communities, for almost all the variables considered over the semantic network, the second and the third most frequent labels are at distance $d - 1$ and $d + 1$ from the *purest* label. It is not the same applying EVA nominal strategy, 3b. The variables for which an ordinal alignment within communities is not reached are *age of acquisition*, *imageability* and *length*: they are not real homophilic properties of a semantic network. This result leads us to study modularity degradation w.r.t. the variables considered. In fact, according to EVA rationale, an alignment between topology and attributes (that might be already present according to mixing pattern distribution) should not considerably

Table 4 The first three columns of the table show EVA results ($\alpha = 0.8$) in the semantic network, where nodes within the same community of *star* are highlighted according to three different dimensions: the semantic domain of *star* according to *size* is quite different from the other two ones. The last column of the table shows LOUVAIN results, where only the topological component is used: a classic approach, contrary to an attribute-aware one, merges in a unique cluster all the meanings of the word

Star clusters				
arousal	**gender**	**size**	Louvain	
dawn	dawn	earth	earth	sky
diamond	gaze	galaxy	night	dark
glitter	glow	infinity	void	abyss
glow	shine	moon	dawn	sunshine
light	sky	planet	celebrity	astronaut
neon	sunrise	sky	sparkle	candle
sparkle	sunset	sun	glitter	diamond
sunshine	world	universe	light	neon

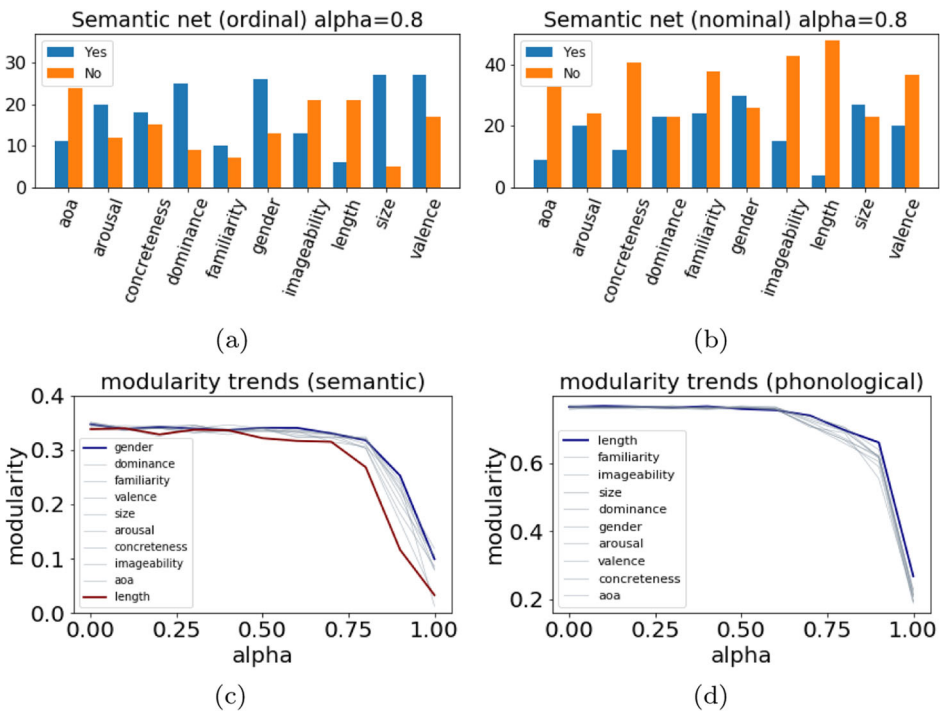


Fig. 3 In the first row, the two bars (respectively, *Yes* or *No*) captures, for each attribute, if the second and the third most frequent labels within a community are at a distance $d \pm 1$ from the most frequent label. The ordinal strategy is applied in **a**, and, in **b**, the nominal one: in **a**, *age of acquisition* (*aoa*), *imageability* and *length* are the dimensions where an ordered label distribution is not achieved in the majority of communities, evidently because these properties do not constitute homophilic dimensions of a semantic network. In the second row, figures **c**-**d**, it is analyzed the modularity degradation w.r.t. the linguistic dimension: in **c** for the semantic network, in **d** in the phonological one

disrupt connectivity in favour of attributes. We tested such a hypothesis in Fig. 3c-d, observing that modularity trends are different w.r.t. the dimensions, and that *age of acquisition*, *imageability* and *length*, in particular, are the variables that obtain the worst results in terms of modularity stability. The results in (a) and (c) are coherent with each other.

In general, we can interpret the modularity decrease as well as its stability as a proxy for the strength of a variable to maintain cluster connectivity if communities are forced to be strongly homogeneous. As expected, *length* is the worst homogeneous dimension in a semantic network, because the length of the words is not a semantic dimension (the same as *imageability* and *age of acquisition*), but *length* is the best one in terms of modularity stability in the phonological layer, where words with a similar sound have also the same or similar length.

EVA as a tool: node label prediction

Finally, in this section, we leverage EVA as a pre-processing step to address relevant network problems: to such extent, we will focus on a classic task, namely node label prediction.

Such task, also known as node classification or node attribute inference, concerns the problem of inferring missing attribute values of nodes, given the values of other nodes.

Several techniques address such task, often involving machine learning and graph embedding methods; since the labeled community discovery task is far from these supervised or semi-supervised techniques directly designed to perform the node prediction task, we approach the issue as a test to validate and exploit EVA outputs, and we limit to a comparison with other similar clustering approaches.

Using the point of view of our proposed clustering method, we model the prediction task as a particular instance of a more general problem:

Definition 6 (**Node label prediction**)**.** Given a partially node-attributed graph $\mathcal{G}$, the task of node label prediction aims to provide high-quality labeling for all nodes carrying missing information.

Given a partially node-attributed graph $\mathcal{G}$ as input, we extend EVA so to make the algorithm resilient to partial information. In practice, while considering an unlabeled node as a candidate for inclusion in a preexisting community, only the structure is used to compare such local choice to the alternative ones. In detail, during the EVA moving phase, nodes with missing information are evaluated only according to the modularity gain (see line 12, Algorithm 2, "The EVA algorithm" section), thus avoiding to consider the purity one.

With this extension, EVA can identify communities whose nodes are partially labeled but still well defined in terms of topology as well as the purity of the labeled nodes they contain. As a result, we can use EVA to reconstruct (i.e., predict) missing node labels by assigning to unlabeled nodes the most frequent label of the community they belong to.

Figure 4 shows the results obtained by EVA on the previously studied *Cora* network in a setup where random removal of node labels is involved. The 3D-plot shows the accuracy score - namely the percentage of nodes labeled by our strategy with the correct original node label - w.r.t. two dimensions, namely the percentage of removed nodes and α. In particular, the figure underlines an overall high accuracy (between 0.8 and 0.7) even if the removal involves the 25% of nodes; then, we do not expect that a meso-scale approach can produce reliable results, even in case of highly homophilic node-label setups.

Figure 5a shows a comparison between EVA and other node clustering strategies when applied to the node label prediction task. Indeed, the alternative strategies tested, namely INFOMAP (Rosvall and Bergstrom 2008), LEIDEN and LOUVAIN, were not explicitly designed (or extended) to address this specific task. As for EVA, the label of each unlabeled node will be the majority one of the community it belongs to. Moreover, we compare the selected approaches against a local baseline, namely the ego-networks decomposition of the original graph: we extract the ego-networks of unlabeled nodes, then we follow the same rationale discussed so far to predict the labels, i.e., using the most frequent value among the direct neighbours. Such a local approach provides a strict boundary to homophilic behaviours, and it is expected to perform well in case of high label assortativity values around unlabeled nodes. Indeed, the obtained results suggest that ego-networks perform better compared to the other approaches, while EVA ranks first among the community discovery ones. We can explain it by observing Fig. 5b, where the local assortativity distribution (Peel et al. 2018) is shown. Newman's assortativity coefficient (Newman 2003) is a global score that describes the whole network without considering outliers and possibly different mixing patterns. Local assortativity computes

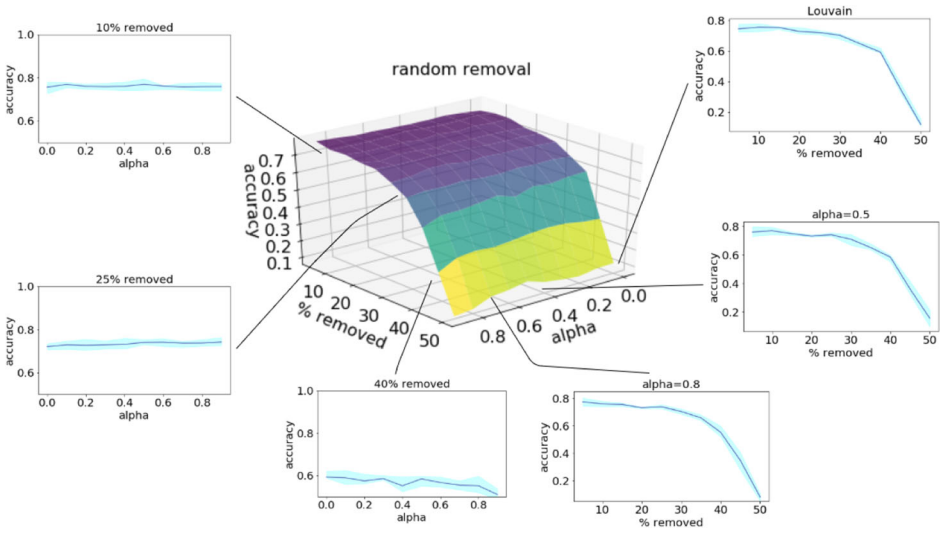


Fig. 4 Node label prediction: random node removal scenario in *Cora* network. Average values of accuracy over 10 executions: around the 3D plot are shown the projections of α vs. *accuracy* and *percentage of nodes removed* vs. *accuracy* (with interquartile ranges)

a homophilic score for each node by considering a local neighbourhood defined by the probability that a random walker with restart reaches a node starting from a target one. In the specific case of Cora, both global assortativity coefficient ($r = 0.76$) and the plotted local one reflect the same results: nodes tend to be highly connected to peers sharing the same label. This specific scenario facilitates local approaches such as the one implemented by ego-networks because the highly homophilic patterns around nodes ensure the reduction of noise when predicting missing labels. It is simpler for the ego-networks to reach a better result compared to the approaches that focus on the discovery of a wider cluster, e.g., the meso-scale network level.

In Fig. 5c-d the prediction task is replicated on a different network, *Web spam*, which presents a different local assortativity distribution. In such a scenario, the predictive power of ego-networks is reduced: the prediction is subject to an increased degree of noise due to the absence of a strongly biased local assortativity distribution peaked towards 1.

Related work

A survey of two main topics is needed to provide a full context for the present work, i) the state-of-art of attribute-aware community discovery techniques and ii) a brief overview of the node label prediction task.

Labeled community discovery. As remarked by a recent survey about the topic (Chunaev 2020), a widely accepted classification schema for labeled community discovery algorithms was taken into little consideration so far. In several comparative studies (Bothorel et al. 2015; Falih et al. 2018), algorithms have been classified according to the different techniques leveraged by their authors to combine topology and semantic levels, usually grouping them in three general families: i) *topological-based*, ii) *attributed-based* and iii) *hybrid* approaches. Such taxonomies focus primarily on *how* performing the clustering step – either i) attaching or ii) merging the attribute information to the topological

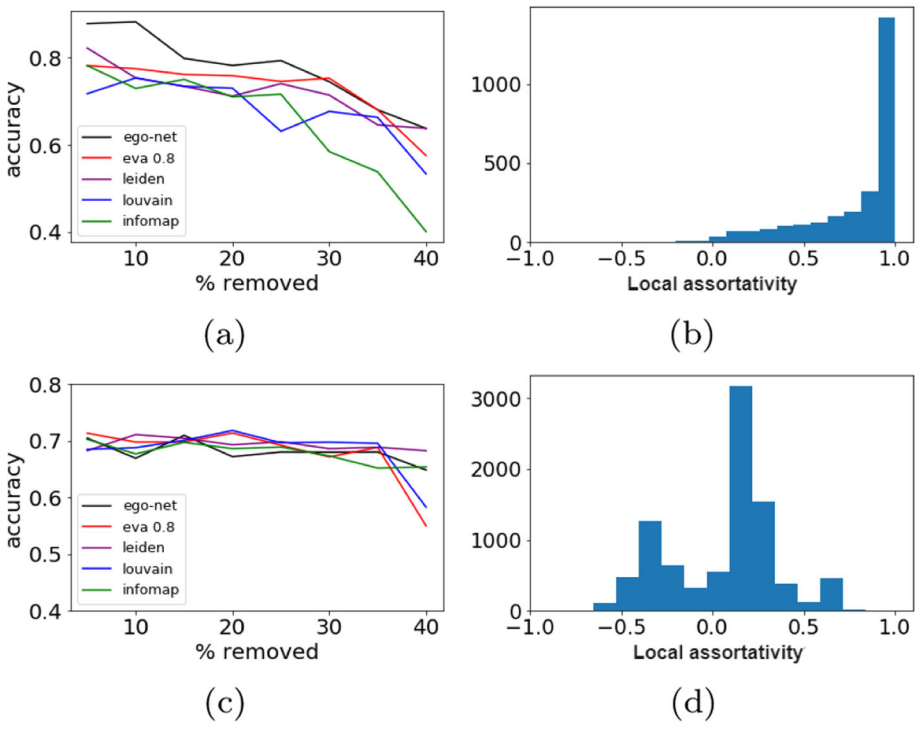


Fig. 5 Prediction accuracy comparisons and network local assortativity distributions: **a**-**b** *Cora*, **c**-**d** *Web spam*

one, or iii) using a posterior ensemble method. Nevertheless, *when* the clustering step is made constitutes the main policy for categorizing algorithms in Chunaev (2020): different methods are grouped according to the moment when structure and attributes are fused (before, during and after the clustering phase), distinguishing between i) *early fusion*, ii) *simultaneous fusion* and iii) *late fusion* methods.

In the following, we review some of the most popular (and interesting to us) labeled community discovery algorithms, while demanding detailed information about them is left to the specific paper they introduce them. SAC2 (Dang and Viennet 2012), STOC (Baroni et al. 2017) and SA-Cluster (Zhou et al. 2009) are *early-fusion* methods. According to the technique used, we can distinguish them between those weighting the edges according to the semantic similarity between nodes (SAC2) or using a node-augmented graph (SA-Cluster) or a distance function (STOC). SA-Cluster is one of the first approaches proposed in the literature. It exploits a graph augmentation technique before the clustering step: a new set of nodes representing the pair *attribute - attribute value* is created, and an edge is added if a *structure node* (i.e., an original node) is labeled with that pair; then, a neighbourhood random walk model is used to measure node proximity. A similar approach is proposed in SAC2, where a k-NN graph is created to connect nodes not only according to the structure. Each node in the k-NN graph is connected to its k most similar nodes, where the similarity is calculated based on a linear combination of structure information (e.g., links) and an attribute similarity function. Thus, a standard community discovery algorithm (e.g., LOUVAIN) is executed on the k-NN graph. STOC was used as a competitor in the current study: in addition to the description given in "Analytical results: single-nominal and multi-nominal" section, STOC is one of the fastest

approaches in literature in terms of time complexity and data-structure employment, and it is able to deal both with categorical and quantitative attributes.

I-Louvain (Combe et al. 2015) and SAC1 (Dang and Viennet 2012) are two examples of *simultaneous fusion* methods. They are the most similar approaches to EVA, because they complement an attribute-aware criterion to modularity, i.e., a semantic distance function in SAC1 (as described in "Analytical results: single-nominal and multi-nominal" section) and a measure called *inertia* in I-Louvain. The I-Louvain approach deals only with quantitative attributes, so it was not possible to compare it with EVA). Probabilistic model-based methods are other *simultaneous fusion* techniques. They assume that structure and attributes are generated according to chosen distributions, translating the attribute-aware community discovery task into a probabilistic inference problem. For instance, BAGC (Xu et al. 2012) adopts a Bayesian criterion to infer the parameters that best fit the graph given in input. CESNA (Yang et al. 2013), built on top of the probabilistic generative process of the BigCLAM overlapping (non attributed) community detection algorithm (Yang and Leskovec 2013), treats node attributes as latent variables.

Finally, another approach is to *switch* between structure-only or attribute-only clustering algorithms according to the clear or ambiguous relations between topology and attributes: it is the rationale behind some *late-fusion* approaches, as the Selection method (Elhadi and Agam 2013), where LOUVAIN (structure selection) or K-Means (MacQueen and et al 1967) (attribute selection) are performed after an evaluation of the boundary between clear and ambiguous structure based on the estimation of an *empirical* mixing parameter μ to compare with the *synthetic* one used in the LFR benchmark (Lancichinetti et al. 2008).

Node label prediction. To show the usefulness of EVA as a tool for accomplishing complex network analysis tasks, in "EVA as a tool: node label prediction" section we leveraged it to predict the label of incomplete nodes. Community discovery is not explicitly designed to perform node classification. Such a task is mainly addressed by graph-based machine learning techniques or approaches within the Statistical Relational Learning framework, or more recently employing embedding methods. As our attribute-aware clustering definition, these methods assume network homophily, but the approaches are different and not easily comparable. For instance, input requests are different. EVA works with labeled datasets as those introduced in this work: a single-categorical scenario or a multi-categorical one where few attributes are considered. The other methods often need a long set of attributes or vectors of features.

For a first distinction, as the one proposed in Bhagat et al. (2011), we mention the existence of methods based on the iterative application of traditional classifiers (a Decision Tree, a Naive Bayes…) (Bhagat et al. 2007) as well as those based on semi-supervised learning (Zhu et al. 2003). The former ones use link information as features; the latter ones exploit information encoded in the link structure for labeling nodes. Another method is to create models characterizing the correlations between the objects that are described by the data. The models within the context of the Statistical Relational Learning are represented by a joint probability distribution of label assignments over nodes of the graph (Taskar et al. 2001): the learnt models are then used to perform inference. More recently, several learning frameworks can achieve successful results using a low-dimension representation of complex networks. The embeddings of nodes are created

from graph topology and from the attributes of the local neighbourhood of each node, as in GraphSAGE (Hamilton et al. 2017).

Conclusion

In this paper, we presented EVA, an attribute-aware node clustering approach that optimizes both topological and attribute-homophilic criteria through a linear combination of modularity and purity. Analytical results show that EVA maintains high modularity scores while identifying clusters of homogeneous labeled nodes – both in single-nominal and in multi-nominal attribute scenarios.

Moreover, EVA can handle ordinal attributes. In such regard, we discussed how our strategy improves state-of-art knowledge about word clustering. We addressed the discovery of semantic/phonological domains in semantic/phonological networks, finding which psycholinguistic properties of words behave as *homophilic glue* in a semantic/phonological network and which do not, because they disrupt modularity trends. We hope these results could be interesting for all those approaches studying language phenomena as complex networks, such as in the emerging field of cognitive network science (Siew et al. 2019), where linguistic systems could be used as a tool to relate language disorders to cognitive impairments.

Finally, we used the node label prediction task to validate EVA performances. We obtained high accuracy scores while exploiting the trade-off between meso-scale connectivity and homogeneity; moreover, our results suggest that different mixing patterns have some effects on the results. Thus, as future works, we plan to properly investigate the relations between homogeneity and assortativity in node-attributed networks. We will aim to design a comprehensive framework able to leverage potentially strict relations between cluster homogeneity and assortative patterns in related complex network analysis tasks.

Abbreviations

EVA: (LOUVAIN) *E*xtended to *V*ertex *A*ttributes; *BESTest*: *B*lockmodel *E*ntropy *S*ignificance *T*est; SAC1: *S*tructural-*A*ttribute *S*imilarities; STOC: *S*emantic-*T*opological *C*lustering; BAGC: *B*ayesian *A*ttributed *G*raph *C*lustering; CESNA: *C*ommunities from *E*dge *S*tructure and *N*ode *A*ttributes; BigCLAM: *C*luster *A*ffiliation *M*odel for *Big* networks; LFR Benchmark: *L*ancichinetti-*F*ortunato-*R*adicchi Benchmark; *GraphSAGE*: Graph *SA*mple and Aggre*GaT*e; *Q* is modularity function; *P* is purity function.

Authors' contributions

Salvatore Citraro and Giulio Rossetti designed research, performed research, and wrote the paper. Salvatore Citraro contributed to the acquisition, the analysis and the interpretation of experiment and tool parts. Giulio Rossetti contributed to the development of the algorithm rationale and its implementation. Both authors read and approved the final manuscript.

Funding

This work is supported by the scheme 'INFRAIA-01-2018-2019: Research and Innovation action', Grant Agreement n. 871042 'SoBigData++: European Integrated Infrastructure for Social Mining and Big Data Analytics'.

Availability of data and materials

- Python code of EVA is available at: https://bit.ly/2J4btLN, and in CDLib (Rossetti et al. 2019);
- Python code of SAC1 is available at: https://github.com/amwat/Graph-Community-Detection;
- Code of STOC is available from the corresponding author on reasonable request;
- The *Sense2vec* network extracted for the analysis is built from *Reddit vectors* available in Kaggle repository, https://www.kaggle.com/poonaml/reddit-vectors-for-sense2vec-spacy/version/1
- The *mental lexicon* network is available from the corresponding author on reasonable request;
- All the other codes and datasets are public and easily available online.

Competing interests

The authors declare that they have no competing interests.

Received: 15 April 2020 Accepted: 4 August 2020
Published online: 26 August 2020

References

Baroni A, Conte A, Patrignani M, Ruggieri S (2017) Efficiently clustering very large attributed graphs. In: Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM). Association for Computing Machinery, New York, NY, USA. pp 369–376. https://doi.org/10.1145/3110025.3110030

Bhagat S, Cormode G, Muthukrishnan S (2011) Node classification in social networks. In: Social Network Data Analytics. pp 115–148

Bhagat S, Rozenbaum I, Cormode G (2007) Applying link-based classification to label blogs. In: Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 Workshop on Web Mining and Social Network Analysis. pp 92–101

Bianconi G, Pin P, Marsili M (2009) Assessing the relevance of node features for network structure. Proc Natl Acad Sci 106(28):11433–11438

Blondel VD, Guillaume J-L, Lambiotte R, Lefebvre E (2008) Fast unfolding of communities in large networks. J Stat Mech Theory Exp 2008(10):10008

Bothorel C, Cruz JD, Magnani M, Micenkova B (2015) Clustering attributed graphs: models, measures and methods. arXiv preprint arXiv:1501.01676

Castillo C, Donato D, Gionis A, Murdock V, Silvestri F (2007) Know your neighbors: Web spam detection using the web topology. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp 423–430

Chunaev P (2020) Community detection in node-attributed social networks: A survey. Comput Sci Rev 37:100286. http://arxiv.org/abs/1912.09816

Citraro S, Rossetti G (2020) Eva: Attribute-aware network segmentation. In: Cherifi H, Gaito S, Mendes JF, Moro E, Rocha LM (eds). Complex Networks and Their Applications VIII. Springer, Cham. pp 141–151

Combe D, Largeron C, Géry M, Egyed-Zsigmond E (2015) I-louvain: An attributed graph clustering method. In: International Symposium on Intelligent Data Analysis. Springer, Cham. pp 181–192

Dang TA, Viennet E (2012) Community detection based on structural and attribute similarities. In: International Conference on Digital Society (ICDS). pp 7–12

Elhadi H, Agam G (2013) Structure and attributes community detection: comparative analysis of composite, ensemble and selection methods. In: Proceedings of the 7th Workshop on Social Network Mining and Analysis. pp 1–7

Falih I, Grozavu N, Kanawati R, Bennani Y (2018) Community detection in attributed network. In: Companion Proceedings of the The Web Conference 2018. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE. pp 1299–1306. https://doi.org/10.1145/3184558.3191570

Fortunato S, Barthelemy M (2007) Resolution limit in community detection. Proc Natl Acad Sci 104(1):36–41

Fortunato S, Hric D (2016) Community detection in networks: A user guide. Phys Rep 659:1–44. https://doi.org/10.1016/j.physrep.2016.09.002

Hamilton W, Ying Z, Leskovec J (2017) Inductive representation learning on large graphs. In: Advances in Neural Information Processing Systems. pp 1024–1034

Interdonato R, Atzmueller M, Gaito S, Kanawati R, Largeron C, Sala A (2019) Feature-rich networks: going beyond complex network topologies. Appl Netw Sci 4(1):1–13. https://doi.org/10.1007/s41109-019-0111-x

Lancichinetti A, Fortunato S, Radicchi F (2008) Benchmark graphs for testing community detection algorithms. Phys Rev E 78(4):046110

Leskovec J, Mcauley JJ (2012) Learning to discover social circles in ego networks. In: Advances in Neural Information Processing Systems. pp 539–547. http://papers.nips.cc/paper/4532-learning-to-discover-social-circles-in-ego-networks.pdf

Likert R (1932) A technique for the measurement of attitudes. Arch Psychol 55:22–140

MacQueen J, et al (1967) Some methods for classification and analysis of multivariate observations. In: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Oakland, CA, USA Vol. 1. pp 281–297

McCallum AK, Nigam K, Rennie J, Seymore K (2000) Automating the construction of internet portals with machine learning. Inf Retr 3(2):127–163

Neville J, Jensen D, Friedland L, Hay M (2003) Learning relational probability trees. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp 625–630

Newman ME (2003) Mixing patterns in networks. Phys Rev E 67(2):026126

Peel L, Delvenne J-C, Lambiotte R (2018) Multiscale mixing patterns in networks. Proc Natl Acad Sci 115(16):4057–4062

Peel L, Larremore DB, Clauset A (2017) The ground truth about metadata and community detection in networks. Sci Adv 3(5):1602548

Rossetti G, Milli L, Cazabet R (2019) CDLIB: a python library to extract, compare and evaluate communities from complex networks. Appl Netw Sci 4(1):52

Rosvall M, Bergstrom CT (2008) Maps of random walks on complex networks reveal community structure. Proc Natl Acad Sci 105(4):1118–1123. https://doi.org/10.1073/pnas.0706851105. http://arxiv.org/abs/https://www.pnas.org/content/105/4/1118.full.pdf

Scott GG, Keitel A, Becirspahic M, Yao B, Sereno SC (2019) The Glasgow Norms: Ratings of 5,500 words on nine scales. Behav Res Methods 51(3):1258–1270

Siew CS, Wulff DU, Beckage NM, Kenett YN (2019) Cognitive network science: A review of research on cognition through the lens of network representations, processes, and dynamics. Complexity 2019

Stella M, Beckage NM, Brede M, De Domenico M (2018) Multiplex model of mental lexicon reveals explosive learning in humans. Sci Rep 8(1):1–11

Taskar B, Segal E, Koller D (2001) Probabilistic classification and clustering in relational data. In: International Joint Conference on Artificial Intelligence. Lawrence Erlbaum Associates LTD Vol. 17. pp 870–878

Traag VA, Waltman L, van Eck NJ (2019) From Louvain to Leiden: guaranteeing well-connected communities. Sci Rep 9(1):1–12. http://arxiv.org/abs/1810.08473

Trask A, Michalak P, Liu J (2015) sense2vec - A fast and accurate method for word sense disambiguation in neural word embeddings. arXiv preprint arXiv:1511.06388

Traud AL, Mucha PJ, Porter MA (2012) Social structure of facebook networks. Physica A: Statistical Mechanics and its Applications 391(16):4165–4180. Elsevier
Xu Z, Ke Y, Wang Y, Cheng H, Cheng J (2012) A model-based approach to attributed graph clustering. In: Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data. pp 505–516
Yang J, Leskovec J (2013) Overlapping community detection at scale: a nonnegative matrix factorization approach. In: Proceedings of the Sixth ACM International Conference on Web Search and Data Mining. pp 587–596
Yang J, McAuley J, Leskovec J (2013) Community detection in networks with node attributes. In: 2013 IEEE 13th International Conference on Data Mining. pp 1151–1156. https://doi.org/10.1109/ICDM.2013.167
Zhou Y, Cheng H, Yu JX (2009) Graph clustering based on structural/attribute similarities. Proc VLDB Endow 2(1):718–729. https://doi.org/10.14778/1687627.1687709
Zhu X, Ghahramani Z, Lafferty JD (2003) Semi-supervised learning using gaussian fields and harmonic functions. In: Proceedings of the 20th International Conference on Machine Learning (ICML-03). pp 912–919

Publisher's Note