Interest clustering coefficient: a new metric for directed networks like Twitter

Thibaud Trolliet, Nathann Cohen, Frédéric Giroire, Luc Hogie, Stéphane Pérennes

Journal of Complex Networks (2021) **00**, 1–30
doi: 10.1093/comnet/cnab030

Interest clustering coefficient: a new metric for directed networks like Twitter

THIBAUD TROLLIET[†]
Inria Sophia-Antipolis, 2004 Route des Lucioles, 06902 Valbonne, France

NATHANN COHEN, FRÉDÉRIC GIROIRE, LUC HOGIE AND STÉPHANE PÉRENNES
Université Côte d'Azur/CNRS, 250 Rue Albert Einstein, 06560 Valbonne, France
[†]Corresponding author. Email: thibaud.trolliet@gmail.com

Edited by: Jesus Gomez-Gardenes

[Received on 1 March 2021; editorial decision on 00 Month 0000; accepted on 6 August 2021]

The clustering coefficient has been introduced to capture the social phenomena that a friend of a friend tends to be my friend. This metric has been widely studied and has shown to be of great interest to describe the characteristics of a social graph. But, the clustering coefficient is originally defined for a graph in which the links are undirected, such as friendship links (Facebook) or professional links (LinkedIn). For a graph in which links are directed from a source of information to a consumer of information, it is no more adequate. We show that former studies have missed much of the information contained in the directed part of such graphs. In this article, we introduce a new metric to measure the clustering of directed social graphs with interest links, namely the interest clustering coefficient. We compute it (exactly and using sampling methods) on a very large social graph, a Twitter snapshot with 505 million users and 23 billion links, as well as other various datasets. We additionally provide the values of the formerly introduced directed and undirected metrics, a first on such a large snapshot. We observe a higher value of the interest clustering coefficient than classic directed clustering coefficients, showing the importance of this metric. By studying the bidirectional edges of the Twitter graph, we also show that the interest clustering coefficient is more adequate to capture the interest part of the graph while classic ones are more adequate to capture the social part. We also introduce a new model able to build random networks with a high value of interest clustering coefficient. We finally discuss the interest of this new metric for link recommendation.

Keywords: complex networks; clustering coefficient; directed networks; social networks; Twitter; link recommendation.

1. Introduction

Networks appear in a large number of complex systems, whether they are social, biological, economical or technological. Examples include neuronal networks, the Internet, financial transactions, online social networks, ... Most 'real-world' networks display some properties that are not due to chance and that are really different from random networks or regular lattices. In this article, we focus on the study of the clustering coefficient of social networks, a metric used to measure the tendency for nodes to form highly connected neighbourhoods. It is classically defined for undirected networks as three times the number of triangles divided by the number of open triangles (formed by two incident edges). This clustering coefficient has been computed in many social networks and has been observed as much higher than what randomness would give. Triangles thus are of crucial interest to understand 'real-world' networks.

However, a large quantity of those networks are in fact directed (e.g. the web, online social networks like Instagram, financial transactions). It is for instance the case of Twitter, one of the largest and most

influential social networks with 126 million daily active users [1]. In Twitter, a person can follow someone she is interested in; the resulting graph, where there is a link $u \to v$ if the account associated to the node u followed the account associated to the node v, is thus directed. In this study, we used as main dataset the snapshot of Twitter (TS in short) extracted by Gabielkov *et al.* as explained in [2] and made available by the authors. The TS has around 505 million nodes and 23 billion arcs, making it one of the biggest snapshots of a social network available today.

The classic definition of the clustering coefficient cannot be directly applied on directed graphs. This is why most of the studies computed it on the so-called *mutual graph*, as defined by Myers *et al.* in [3], that is, on the subgraph built with only the bi-directional links. We call *mutual clustering coefficient (mcc for short)* the clustering coefficient associated with this graph. We computed this coefficient in the TS, using both exact and approximated methods. We find a value of 10.7%, a high value of the same order than the ones found in other web social networks [4, 5].

However, this classical way to operate *leaves out 2/3 of the graph!* Indeed, the bi-directional edges only represent 35% of the edges of the TS. A way to avoid it is to consider all links as undirected and to compute the clustering coefficient of the obtained undirected graph. We call *undirected clustering coefficient (ucc for short)* the corresponding computed coefficient. Such a computation in the TS gives a value of ucc of only 0.11%. This is way lower than what was found in most undirected social networks. It is thus a necessity to introduce specific clustering coefficients for the directed graphs. More generally, when analysing any directed datasets, it is of crucial importance to take into account the information contained in its directed part in the most adequate way.

A first way to do that is to look at the different ways to form triangles with directed edges. Fagiolo computed the expected values of clustering coefficients considering directed triangles for random graphs in [6] and illustrated his method on empirical data on world-trade flows. There are two possible orientations of triangles: transitive and cyclic triangles, see Fig. 1b and c. Each type of triangles corresponds to a directed clustering coefficient :

- the *transitive clustering coefficient* (*tcc* in short), defined as:

$$tcc = \frac{\text{No. of transitive triangles}}{\text{No. of open transitive triangles}},$$

- the *cyclic clustering coefficient* (*ccc* in short), defined as:

$$ccc = \frac{3 \cdot \text{No. of cyclic triangles}}{\text{No. of open transitive triangles}}.$$

We computed both coefficients for the snapshot, obtaining $tcc = 1.9\%$ and $ccc = 1.7\%$. However, note that a large part of the transitive and cyclic triangles comes from bi-directional triangles. When removing them, we arrive to values of $tcc = 0.51\%$ and $ccc = 0.24\%$.

We believe those metrics miss an essential aspect of the Twitter graph: while the clustering coefficient was defined to represent the social cliques between people, it is not adequate to capture the information aspect of Twitter, known to be both a social and information media [3, 7]. In this work, we go one step further in the way directed relationships are modeled. We argue that in directed networks, *the best way to define a relation or similarity between two individuals (Bob and Alice) is not always by a direct link, but by a common interest*, that is, two links towards the same node (e.g. Bob $\to$ Carol and Alice $\to$ Carol). Indeed, consider two nodes having similar interests. Apart from being friends, these two nodes

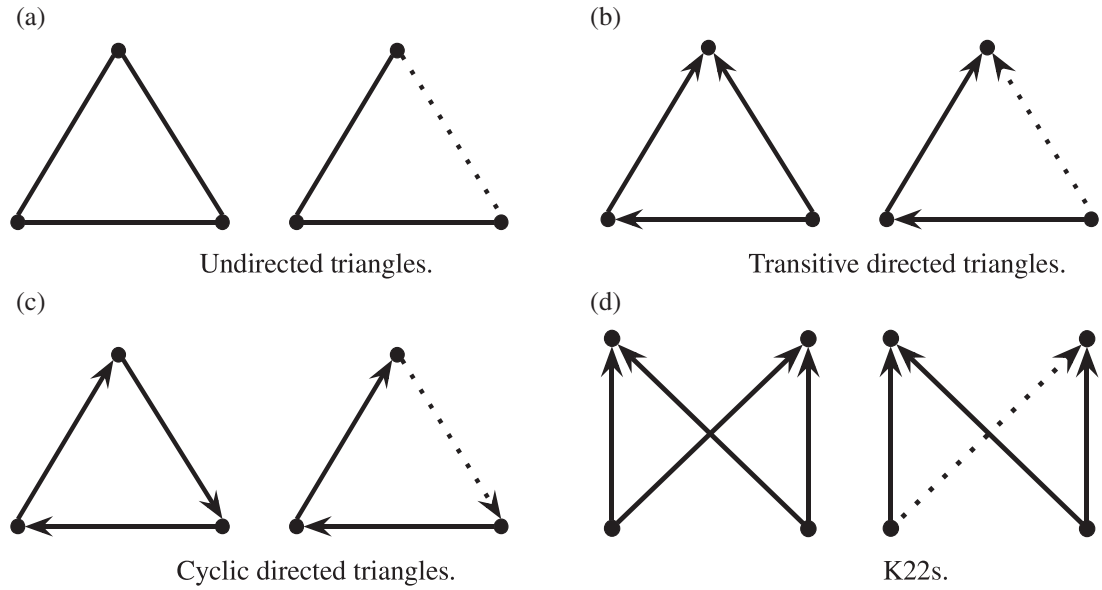

FIG. 1. Closed (left) and open (right) undirected and directed triangles and K22s. (a) Undirected triangles. (b) Transitive directed triangles. (c) Cyclic directed triangles. (d) K22s.

do not have any reason to be directly connected. However, they would tend to be connected to the same out-neighbours. We exploit this to study a new notion of connections in directed networks and the new naturally associated clustering coefficient, which we name *interest clustering coefficient, or icc* in short. We define it as follows:

$$icc = \frac{4 \cdot \#\,\text{K22s}}{\#\,\text{open K22s}},$$

where a K22 [1] is defined as a set of four nodes in which two of them follow the two others, and an open K22 is a K22 with a missing link, see Fig. 1d. We computed the icc on the Twitter snapshot, obtaining a value of 3.6% (3.1% when removing the bi-directional structures). This value, an order of magnitude higher than the previous clustering coefficients computed on the non-bidirectional directed graph, confirms the interest of this metric. If the clustering coefficients of triangles are good metrics to capture the social aspect of a graph, the interest clustering coefficient is a good metric to capture the informational aspect.

In summary, our contributions are the following:

- We define a new clustering coefficient for graphs with interest links.
- We succeeded in computing it, both exactly and using sampling methods, for a snapshot of Twitter with 505 million nodes and 23 billion edges.

[0] The name comes from Graph Theory. A $K_{m,n}$ is a complete bipartite graph $G = (V_1 \cup V_2, E)$ with partitions of sizes $|V_1| = m$ and $|V_2| = n$. We consider in this article a directed version of a $K_{2,2}$.

- We additionally provide the values of the directed and undirected clustering coefficients previously defined in the literature. We believe this is the first time that such coefficients are computed exactly for a large *directed* online social network.
- We compute this new metric as much as the previous ones on other directed datasets to highlight the differences and interests of the different metrics.
- We then propose a new random graph model to obtain random directed graphs with a high interest clustering coefficient. We prove this model follows power-law in- and out-degree distributions, and analyse the interest clustering coefficient value by simulation.
- Lastly, we discuss the usage of this new metric for link recommendation. The principle is to recommend links closing a large number of K22s (instead, classically, of triangles). We discuss the strengths/weaknesses of this method for a set of Twitter users.

The article is organized as follows. We first discuss related work in Section 2. In Section 3, we present the algorithms we used to compute the values of the interest clustering coefficient, both exactly and by sampling. We discuss the results on the clustering coefficients of Twitter in Section 4, and of other directed datasets in Section 5. In Section 6, we propose and study a preferential attachment model providing a high interest clustering coefficient. Lastly, we discuss the use of interest clustering coefficient for link recommendation in Section 7.

2. Related work

2.1 *Complex networks*

Even if the study of complex networks is an old field [8], it keeps receiving a lot of attention from the research community. The reason for this is two-fold. First, a great number of very large practical systems which emerged recently can be seen as complex networks, in particular online social media networks, see [9] for a survey. Second, with the development of big data analysis, entrepreneurs, analysts or researchers have new tools to study those huge amounts of data. Complex networks often share common properties, like small diameter [10], small average distance [11–13], heavy tail degree distributions [12, 14], high clustering [13], communities [5], etc.

2.2 *Clustering coefficient*

Among those properties, the clustering coefficient shows that, when two people know each other, there is a high probability that those people have common friends. The clustering coefficient has numerous important applications, such as spam detection [15], link recommendation [16, 17], information spread [18], study of biased network samples [19], performance of some neural networks [20], etc. There are different definitions of the clustering coefficient. The *local clustering coefficient* of a node i, first introduced by Watts and Strogatz [13], is defined as the probability that two neighbours of i are also connected together. This probability can be computed as

$$CC(i) = \frac{\text{No. of triangles with the node i}}{\text{No. of connected triplets centred on i}},$$

where (No. of connected triplets centred on i) = $\binom{\deg(i)}{2}$. From here a clustering coefficient can be defined for the whole graph as the mean of the local clustering coefficients over all the nodes of the graph:

$$CC_{g1} = \frac{1}{n} \sum_{i \in V} CC(i)$$

Another definition was first introduced by Barrat and Weigt in [21], and is called the *global clustering coefficient*, or *transitivity*. It is defined as

$$CC_g = 3 \times \frac{\text{No. of triangles in the graph}}{\text{No. of connected triplets of vertices in the graph}}.$$

We use the global clustering coefficient in this article. The clustering coefficient has also been defined for weighted graphs [22, 23].

2.3 *Computations for social graphs*

The undirected clustering coefficient of some social networks has been provided in the literature. It has been computed on very large snapshots for Facebook [5], Microsoft Messenger [12], Flickr and YouTube [4]. The local clustering coefficient has also been studied in the undirected mutual graph of Twitter [3]. We can also cite the values given by the Network Repository project [24], providing a large comprehensive collection of network graph data available for which it lists some basic properties. The undirected clustering coefficient is usually much higher in social networks than in random models.

2.4 *Directed graphs*

All these studies only consider the undirected clustering coefficient, even for directed networks like Twitter. Fagiolo introduced definitions of directed clustering coefficients, that we named tcc and ccc [6], but those definitions have never been computed and discussed on large datasets to our knowledge, as we do in this article. Moreover, we believe that these metrics are *not the most relevant ones for directed graphs with interest links.*

2.5 *Computing sub-structures*

Researchers studied methods to efficiently compute the number of triangles in a graph, as naive methods are computationally very expensive on large graphs. Two families of methods have been proposed: triangle exact counting or enumeration and estimations. In the first family, the fastest algorithm is due to Alon *et al.* [25] and runs in $O(m^{\frac{2\omega}{\omega+1}})$, with m the number of edges and ω the best known exponent for the fast matrix multiplication. Its current value is 2.3728, due to an algorithm of [26] improved by [27], giving a complexity of $O(m^{1.41})$ for the AYZ algorithm. However, methods using matrix multiplication cannot be used for large graphs because of their memory requirements. In practice, enumeration methods are often used, see for example [28, 29]. A large number of methods for approximate counting were proposed, see for example [30] and its references. The authors obtain a running time of $O(m + \frac{m^{3/2} \log n}{t\varepsilon^2})$ and a $(1 \pm \varepsilon)$ approximation. Methods to count rectangles and butterfly structures in undirected bipartite networks were also proposed in [31] and in [32]. In this article, we propose an efficient enumeration algorithm to count

the number of K22s and open K22s in a very large graph. We focused on the case in which only one adjacency can be stored, as this was our case for the TS. To the best of our knowledge, we are the first to consider this setting.

3. Computing clustering coefficients in Twitter

We computed the interest clustering coefficient and the triangle clustering coefficients on a directed Twitter snapshot (TS in short) that we use as a typical example of a directed social network with interest links. We used two different methods: an exact count and an estimation using sampling techniques, either with a Monte Carlo algorithm or with a sampling of the graph.

3.1 *The Twitter snapshot*

In order to compute the different clustering coefficients of a real graph, the authors of [33] gave us access to a snapshot of the graph of the followings of Twitter. The snapshot was collected between March 2012 and July 2012. With $n = 505$ million nodes and $m = 24$ billion links collected, this graph is the largest directed social network graph available today, to the best of our knowledge. Each node of the graph represents an account of Twitter, and there is a link between two nodes u and v if the account u follows the account v. All account IDs have been anonymized. The snapshot is a perfect case study as Twitter is a directed social network used both as a social and an information network [3, 7]. It allows to study directed/undirected social/interest clustering coefficients.

Degree distributions of the Twitter Snapshot. We provide in Fig. 2 the degree distributions of the TS. We fitted their tails to power law distributions. We obtained $P^-(i) = C^- i^{-2.17}$ and $P^+(i) = C^+ i^{-2.76}$, with $P^-(i)$ (respectively $P^+(i)$) the probability that a node has in-degree (respectively out-degree) i. In the following, we use the obtained values to compute the practical complexity of the algorithms.
Other references of the literature have also provided a power law fit for both distributions, see for example [3]. In this work, the authors obtained exponents of values 1.35 and 1.28. However, we believe that the authors did a fit on the complete distributions and not on their tails, leading to power law exponents below 2. This is why we preferred to only fit the tail. Another point of discussion would be to decide if the

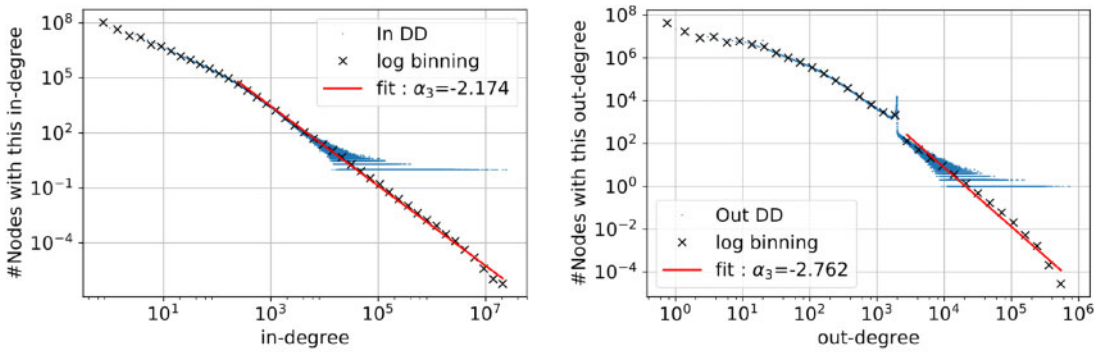

FIG. 2. In- (left) and out-degree (right) distributions of the Twitter Snapshot. The obtained distribution is given by the blue points; the black crosses represent the logarithmic binning of the distribution (a mean of a given amount of points on a logarithmic scale). The red straight line is the fit of the logarithmic binning; it has slopes of -2.174 and -2.762 for the in and out degree distribution.

out-degree distribution really behaves as a power law. However, the best fit of the distributions is out of the scope of this article. We just used the values provided by our fit as a possible model of the graph, but others exist.

3.2 *Exact count*

We computed the exact numbers of K22s and open K22s in the Twitter Snapshot. Recall that we are discussing a dataset with hundreds of million nodes and billions of arcs. Results are reported in Table 1 and discussed in Section 4. We also retrieved the number of directed and undirected triangles of TS. We first discuss the complexity of algorithms for exact counting on very large graphs. We then present the algorithms we use and discuss the results.
In the rest of this article, we call *top vertices* (respectively *bottom vertices*) of a K22 the vertices which are destinations (respectively sources) of the K22 edges. We call a *fork* a set of two edges of a K22 connected to the same vertex. We say that a *fork has top (or bottom) vertex x* if both edges are connected to x and x is a top (respectively bottom) vertex of the K22. The same terminology applies to open K22s.

Trivial algorithm. The trivial algorithm would consider all quadruplets of vertices with two upper vertices. Then, for each quadruplet, it would check the existence of a K22 and of open K22s. There are $\binom{4}{2}\binom{n}{4}$ such quadruplets. It thus gives a complexity of $O(n^4)$. This method can thus not be considered for the TS as it would perform 6.4×10^{22} iterations.

Improved algorithm. The practical complexity can be greatly improved by only considering *connected quadruplets*, and by mutualizing the computations of the common neighbours of the in-neighbours of a vertex, as explained below. The pseudo-code is given in Algorithm 1.
The algorithm main loop iterates on the vertices of the graph. For each vertex x, we consider its in-neighbourhood $N^-(x)$. We then compute how many times a vertex w (with $w < x$ to avoid counting a K22 twice) appears in the out-neighbourhoods of the vertices of $N^-(x)$. We denote it $\#occ(w)$. We use a hash table to store the value of $\#occ(w)$ in order to be able to do a single pass on each out-neighbour.
For a vertex w, any pair of its $\#occ(w)$ in-neighbours common with x forms a K22 with x and w as top vertices. There are hence $\binom{k}{2}$ K22s with x and w as top vertices. The number of K22s with x as a top vertex then is

$$\#K22(x) = \sum_{w|\#occ(w)\geq 2} \binom{\#occ(w)}{2}.$$

The number of open K22s with x as the top vertex is computed by noticing that, for any pair of vertices u and v of $N^-(x)$, we have $d^+(u) - 1 + d^+(v) - 1 - \mathbb{1}_{v\in N^+(u)} - \mathbb{1}_{u\in N^+(v)}$ open K22s containing this fork (ux, vx). We can count the number of open K22s with x as a top vertex, u as the bottom vertex of out-degree 2 (and thus another vertex v as the bottom vertex of out-degree 1). A vertex $u \in N^-(x)$ thus is in $(d^+(u) - 1 \sum_{v\in N^-(x)\setminus\{u\}} 1_{v\in N^+(u)})(d^-(x) - 1)$ such open K22s. The only subtlety is that we count the number of arcs, which are between two vertices of $N^-(x)$, during the loop on the out-neighbourhoods of the vertices of $N^-(x)$. We note this number #internalArcs. We then have:

$$\#openK22(x) = \left(\sum_{u\in N^-(x)} (d^+(u) - 1)(d^-(x) - 1)\right) - \#internalArcs.$$

Lastly, the global number of K22s (respectively open K22s) in the digraph is simply the sum over all vertices x of the number of K22s (respectively open K22s) with x as a top vertex. Note that, since we only consider K22s formed with a vertex w such that $x < w$, we only count each K22 once.

Algorithm 1 Enumeration of K22s and open K22s

```
1. ▷
2. Input: Digraph(V, A)
3. #occ=0                                                    ▷ hash table
4. for x ∈ V do
5.     #internalArcs ← 0   ▷ We count the number of arcs internal to N⁻(x) as these arcs do not form
   open K22s
6.     for v ∈ N⁻(x) do
7.         #openK22s += (d⁺(v) − 1)(d⁻(x) − 1)
8.         for w ∈ N⁺(v) \ {x} do
9.             #occ[w] += 1
10.            if w ∈ N⁻(x) then          ▷ We use a second hash table to test that.
11.                #internalArcs += 1
12.    for w with #occ[w] ≥ 2 do
13.        #k22 += (#occ[w] choose 2)
14.    #openK22s −= #internalArcs
15.    #occ ← 0                                              ▷ Done with a double loop
16. icc ← 4#K22 / #openK22
```

Complexity of the used algorithm. The complexity thus is $m + \sum_u d^+(u)(d^+(u) - 1)$. Indeed, each edge is only considered once as an in-arc and $d^+ - 1$ times as an out-arc. Note that, in the Twitter Snapshot, the sum of the squares of the degrees is equal to 8×10^{13}. The order of the number of iterations needed to compute the number of K22s was thus massively decreased from the 6.4×10^{33} iterations of the trivial algorithm.

Complexity on graphs following a power-law degree distribution. The complexity of the algorithm on a graph built with preferential attachment can be computed as follows. We consider without loss of generality that the sum of the square of the degrees is minimum for the out-degrees (and not the in-degrees). The maximum degree is $d^+_{\max} = O(n^{1/(\alpha^+ - 1)})$, with α^+ the exponent of the out-degree power law distribution. Thus, the sum of the squares of the degrees, when $2 \leq \alpha^+ < 3$, is $\sum_{v \in V}(d^+(v))^2 = C^+ n \sum_{i=1}^{d^+_{\max}} \frac{i^2}{i^{\alpha^+}} \underset{n \to \infty}{\sim} C^+ n \int_{i=1}^{d^+_{\max}} \frac{1}{i^{\alpha^+ - 2}} = \left[\frac{C^+ n}{(3-\alpha^+) i^{\alpha^+ - 3}} \right]_1^{d^+_{\max}} \simeq \frac{C^+ n}{(3-\alpha^+) {d^+_{\max}}^{\alpha^+ - 3}} = \frac{C^+}{(3-\alpha^+)} n^{1 + \frac{3-\alpha^+}{\alpha^+ - 1}}$, where $C^+ = \frac{1}{\sum_{i \in \mathbb{N}^+} i^{\alpha^+}}$. The complexity is thus in $O(m + n^{1 + \frac{3-\alpha^+}{\alpha^+ - 1}})$. For preferential attachment graphs with exponents between 2 and 3, this gives a complexity between $O(m+n)$ and $O(n^2)$, to be compared to the one of the naive method $O(n^4)$.

Counting the number of triangles. The number of transitive triangles can easily be computed for free while counting the K22s. When iterating over the vertices of the TS and considering the vertex x in Algorithm 1, the number `internal_arcs` of arcs between vertices of $N^-(x)$ corresponds to the number of transitive triangles for which x is the top vertex. The number of open transitive triangles with x

as the top vertex is simply $d^-(x) \cdot d^+(x)$. The total number of open transitive triangles is then just the sum of this quantity over all x. The number of cyclic triangles for x can also be easily computed by counting the number of arcs from $N^+(x)$ to $N^-(x)$. Each cyclic triangle is counted three times. The number of open cyclic triangles is the same as the number of open transitive triangles. We can compute the number of undirected triangles with similar methods (either on the full (but undirected) graph or on the mutual graph). Note that the fastest methods to compute triangles in graphs have a complexity of $O(m^{1.41})$, where m is the number of edges [25]. These methods rely on fast matrix multiplications and cannot be applied for large graphs as they need to have the full matrix in memory. Moreover, our algorithms would be faster in practice for large complex networks as they are sparse graphs. The average indegree (or outdegree) has a low value of 45.6 [2] in Twitter. The complexity of the matrix methods would be of the order of 3.2×10^{14} for the TS as $m = 2.3 \times 10^{10}$. This is higher than the practical complexity of computing the exact number of K22s (which is itself higher than the complexity of computing triangles). We discuss the obtained results with the exact count in Section 4.

3.3 *Approximate counts*

As discussed later in Section 4, the exact count of the number of K22s and open K22s in Twitter implies massive computations. This number can be estimated using Monte Carlo Method and/or computations on a sample of the graph. We discuss both methods below. One of our goals was to see how good computations made in the literature using smaller Twitter snapshots were.

3.3.1 *Exact icc on Twitter samples.* We built samples of the TS to estimate the interest clustering coefficient. Several choices can be made to build the samples. To avoid missing nodes of high degrees (which would lead to a high variance), we sampled the arcs (and not the nodes). Given a sampling probability p, we keep an arc in the sample with probability p. We generated samples of different sizes corresponding to sampling probabilities from $p = 1/100$ to $p = 1/16000$.

Estimator of the number of K22 and open K22s. Let us call $\mathcal{A}$ the set of occurrences of a specific pattern (in our case, either a K22 or an open K22). The number of occurrences of the pattern in a sample, X, is given by $X = \sum_{A \in \mathcal{A}} X_A$, where X_A is the random variable which is equal to 1 if all the arcs of pattern A are selected in the sample and 0 otherwise.

If we note l the number of arcs of the pattern (4 for a K22 and 3 for an open K22), we have that $\mathcal{P}[X_A = 1] = p^l$. By linearity of the expectation, we get $\mathbb{E}[X] = p^l|\mathcal{A}|$. Thus, $Y = p^{-l}X$ is an unbiased estimator of $|\mathcal{A}|$.

Variance. Note that the random variables X_A are not independent, that is, two K22s can share a common link. Otherwise, the variance would simply be $\mathbb{V}(X) = \sum_{A \in \mathcal{A}} \mathbb{V}[X_A] = |\mathcal{A}|p^l(1-p^l) \leq |\mathcal{A}|p^l$. However, we can argue that (and we will verify that), in practice, most of the K22s and open K22s do not share any link. It can be used in the analysis as follows.

$$\mathbb{V}[X] = \mathbb{E}[X^2] - \mathbb{E}[X]^2 = \mathbb{E}[(\sum_{A \in \mathcal{A}} X_A)^2] - \mathbb{E}[X]^2 \tag{3.1}$$

$$= \sum_{(A,B) \in \mathcal{A}} \mathbb{E}[X_A X_B] - \mathbb{E}[X]^2. \tag{3.2}$$

We now distinguish the couples of dependent patterns, which we note $\Delta = \{(A, B) \mid A \cap B \neq \emptyset\}$, from the couples of independent ones, $\bar{\Delta} = \{(A, B) \mid A \cap B = \emptyset\}$.

$$\mathbb{V}[X] = \sum_{(A,B)\in\bar{\Delta}} \mathbb{E}[X_A X_B] + \sum_{(A,B)\in\Delta} \mathbb{E}[X_A X_B] - \mathbb{E}[X]^2. \tag{3.3}$$

When A and B are independent, we have

$$\mathbb{E}[X_A X_B] = \mathbb{E}[X_A]\mathbb{E}[X_B] = p^{2l}.$$

As $\mathbb{E}[X]^2 = p^{2l}|\mathcal{A}|^2$, we get

$$\mathbb{V}[X] = \sum_{(A,B)\in\bar{\Delta}} \mathbb{E}[X_A]\mathbb{E}[X_B] + \sum_{(A,B)\in\Delta} \mathbb{E}[X_A X_B] - \mathbb{E}[X]^2 \tag{3.4}$$

$$= \sum_{(A,B)\in\Delta} (\mathbb{E}[X_A X_B] - p^{2l}) \tag{3.5}$$

Let us now distinguish different cases. We note Δ_i the set of couples of patterns sharing $1 \leq i \leq l$ arcs. For a couple $(A, B) \in \Delta_i$, we have that $\mathbb{E}[X_A X_B] = p^{2l-i}$, giving that

$$\mathbb{V}[X] \leq \sum_{i=1}^{l} \sum_{(A,B)\in\Delta_i} (p^{2l-i} - p^{2l}). \tag{3.6}$$

Since $p < 1$, we get

$$\mathbb{V}[X] \leq \sum_{i=1}^{l} p^{2l-i} \left|\Delta_i\right|. \tag{3.7}$$

Note that, when all patterns are independent, $|\Delta| = |\Delta_l| = |\mathcal{A}|$ (couples $(A, A) \in \mathcal{A}$), giving back the variance of the independent case, $p^l|\mathcal{A}|$. Chebycheff's inequality tells us that:

$$Prob[|X - \mu| \geq k\sigma] \leq \frac{1}{k^2}, \tag{3.8}$$

where μ is the expectation and σ is the standard deviation of X. In our case, if we want an accuracy of ε with a probability q, we should have $\frac{1}{k^2} \leq 1 - q$ and $k\sigma \leq \varepsilon p^l|\mathcal{A}|$, which can be rewritten as:

$$\frac{k^2}{\epsilon^2} \sum_{i=1}^{l} p^{2l-i} \frac{|\Delta_i|}{|\mathcal{A}|^2} \leq p^{2l}. \tag{3.9}$$

Lastly, to estimate the icc, we use as an estimator

$$Z = \frac{4Y}{Y_0}, \tag{3.10}$$

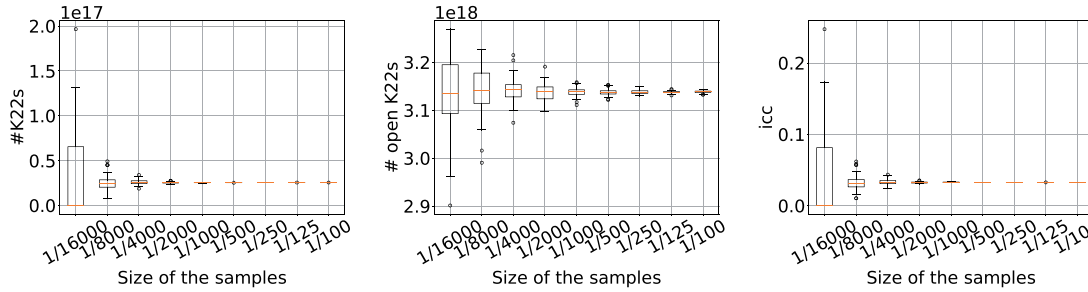

FIG. 3. Estimation of the K22s (left), open K22s (middle) and interest clustering coefficient (right) for different sample sizes.

with Y and Y_o the estimators of the number of K22s and open K22s, respectively. As $\lim_{n\to\infty} Y = \#K22s$ and $\lim_{n\to\infty} Y = \#openK22s$ with n the size of the graph, we have that $\lim_{n\to\infty} Z = icc$. For the precision, if Y and Y_o have an accuracy of ε and ε_o respectively, then with a probability $q = 0.99$, Z has at least an accuracy of $\frac{1+\varepsilon}{1-\varepsilon_o} \underset{\varepsilon\to 0}{\sim} 1 + \varepsilon + \varepsilon_o$ with a probability $q^2 \approx 0.98$.

Numerical application. We now consider the K22s of the TS. Note that we know that $\frac{|\Delta_4|}{|\mathcal{A}|^2} = 1/\#K22s = 3.8 \times 10^{17}$. We also can notice that $|\Delta_3| = |\Delta_4|$. In the TS, an edge is shared by $\frac{\#K22s}{m}$ K22s on average, with m the number of links of the TS. Thus, the average number of K22s sharing at least an edge with a K22 is between $\frac{\#K22s}{m}$ and $4 \cdot \frac{\#K22s}{m}$. It gives $\frac{1}{m}|\mathcal{A}|^2 \leq \Delta_1 + \Delta_2 + \Delta_3 + \Delta_4 \leq \frac{4}{m}|\mathcal{A}|^2$. The number of overlapping K22s with i arcs is a non-increasing function of i. To make a numerical evaluation, we suppose that most overlapping K22s share one edge and not 2 edges in the TS. We set that $|\Delta_1| = \frac{1}{m} = 4.3 \times 10^{-8}|\mathcal{A}|^2$, and $|\Delta_2| = 10^{-16}|\mathcal{A}|^2$. Now, if we want a precision of $\varepsilon = 0.1$ with a probability 0.99 (that is $k = 10$), we need to take a sampling probability p such that

$$p^8 \geq \frac{10^2}{10^{-4}}(p^7 4.3 \times 10^{-8} + p^6 \times 10^{-16} + p^5 3.8 \times 10^{-17} + p^4 3.8 \times 10^{-17}). \tag{3.11}$$

Equation 3.11 is valid for $p \geq 2.5 \times 10^{-4}$. Thus, under these hypotheses, a sample with sampling probability 1/2500 and larger, for example, our 1/2000 sample, allows to estimate the number of K22s with a precision of 10%. The number of open K22s is larger and thus, the precision is better. It gives a precision of at least $\frac{1+1/100}{1-1/100} = 0.20$ for the estimation of the icc. In practice, the Chebysheff inequality and our hypothesis are pessimistic as shown below.

Results. We present in Fig. 3 the results of the algorithm for different sample sizes, corresponding to sampling probabilities from $p = 1/100$ to p=1/16000. For each sample size, we generated 30 samples. The distribution over the samples of the interest clustering coefficient, K22s and open K22s are provided by a boxplot for each value of p. Note that a K22 of the TS appears in a sample with a probability of only p^4, and of p^3 for an open K22. The clustering coefficient of a sample is thus an estimate of $p \cdot icc$.

We observe that the clustering coefficient is well estimated using any sample for a sampling probability of $1/1000$ or larger. Indeed, for this range of probabilities, the distribution over all samples is very concentrated and around the exact value of the icc. Note that, for $p = 1/1000$, a K22 is present in the sample with a probability of only 10^{-12}. The expectation of the number of nodes with an edge is only 23 million nodes (over 500 million) and the number of edges is also around 23 million (over 23 billion). Thus, a small sample (5% of the nodes and 0.1% of edges) allows to do an efficient estimation of the icc.

For smaller values of p, the variance increases. The median estimates well the icc for a range of

p between 1/8000 and 1/1000, but samples of these sizes may have error of 100% of the value. Lastly, for $p = 1/16000$, only the number of open K22s (and not the K22s or the icc) is approximated by the median.
In conclusion, a sample with sampling probability 1/1000 is enough to efficiently estimate the interest clustering coefficient, with a computation time of around 1 min (instead of days for the whole TS) on a machine of the cluster.

3.3.2 *Monte Carlo method* After a short reminder of the precision of the Monte Carlo method, we first quickly discuss the case of triangles to show the particularity of estimating the interest clustering coefficient. The difficulty here is that the probability to observe a (closed or open) K22 or a triangle is very small. In the case of triangles, this difficulty can be easily circumvented by knowing the node degrees. This allows to select an open triangle uniformly at random. In the case of K22s, this information is not sufficient to select an open K22 uniformly at random. In fact, achieving this goal is very costly, but we present a method in which, by picking only forks (as we do for triangles), we can compute the interest clustering coefficient.

Preliminary: precision of Monte Carlo method. *Precision of the estimation and number of iterations.* Each trial is a Bernoulli variable with probability p. We use as an estimate Y, the mean of the random sample. Its expectation is p and its standard deviation is $\frac{\sqrt{p(1-p)}}{\sqrt{n}}$. Due to the central limit theorem, we get that, when n is large,

$$Prob\left[|Y-p| \leq Z_{\alpha/2}\frac{\sqrt{p(1-p)}}{\sqrt{n}}\right] = \alpha, \tag{3.12}$$

with $Z_{\alpha/2}$ the value giving the α confidence interval a standard normal distribution. To get with probability α an accuracy of ε of the empirical mean p (which is not known), we should have $Z_{\alpha/2}\frac{\sqrt{p(1-p)}}{\sqrt{n}} \leq \varepsilon p$. That is $n \geq \frac{Z^2_{\alpha/2}(1-p)}{p\varepsilon^2}$. If we take $n \geq \frac{Z^2_{\alpha/2}}{p\varepsilon^2}$, we have the wanted precision (and we are not doing many more iterations when p is small). For example, to get an accuracy of 99% ($\varepsilon = 0.01$), with probability $\alpha = 0.99$, we should have a number of iterations such that $n \geq \frac{75{,}625}{p}$.

Approximating the number of undirected triangles. A first direct method would be to select three vertices uniformly at random and check if they form a triangle and open triangles. The problem with this method is that the probability to form a triangle in Twitter is the number of triangles divided by the number of triplet of nodes, i.e., $\frac{6.23\times10^{11}}{(5\times10^8)^3} = 5\times10^{-15}$. Thus the number of needed iterations would be astronomic, 5.5×10^{19} for an accuracy of 1%, with probability $\alpha = 0.99$. We thus have to use methods selecting open triangles directly.

To estimate the undirected clustering coefficient, we need to select open (undirected) triangles uniformly at random. We then test if the selected triangle is closed or not (which is the case with probability ucc). The number of open triangles rooted at vertex v is equal to $\frac{d(v)d(v)-1}{2}$. We can thus perform the sampling by picking a vertex v with probability $\binom{d(v)}{2}/\sum_{v\in V}\binom{d(v)}{2}$ and then select two random edges adjacent to v.

Directed triangles. The method is the same in the case of directed triangles. We select an open triangle uniformly at random. The number of open triangles rooted at a vertex v is $d^-(u)d^+(u)$. We thus select a node u with probability $d^-(u)d^+(u)/\sum_{v\in V} d^-(v)d^+(v)$. We then select uniformly at random an incoming

arc and an outgoing arc. Lastly, we check if the triangle is closed (which is the case with a probability equal to tcc and to ccc respectively for transitive and cyclic triangles).
Precision of the estimation and number of iterations. Each trial is a Bernoulli variable with a probability $p = tcc = 0.019$. To get an accuracy of 1%, with probability 0.99, we should thus do $n = 4 \times 10^6$ iterations.

Interest clustering coefficient. For triangles, we were able to select uniformly at random open triangles using the node degrees. In the case of K22s, node degrees is not sufficient to select an open K22 uniformly at random. To do so, it would be necessary to compute the number of open K22s with u as a root. This pre-processing is very costly: for each node, we should consider its in-neighbours, sum their out-degrees, and compute the number of internal edges. It would be almost as costly as doing an exact count of the number of K22s.

Another method is to select a vertex v as a root according to the square of its in-degree (as in the case of triangles), but without knowing its number of open K22s (first step). We then select two arcs u_1v and u_2v uniformly at random (second step). We then compute the number of K22s and open K22s with the selected fork (u_1v, u_2v) (third step).

For the first step, the algorithm needs a list of the node in-degrees of the TS, which would have been computed in a preliminary step. For the second one, it then uses the in-adjacency of v. For the third step, the out-adjacency of u_1 and u_2 are necessary for the computations.

We then use the estimators introduced below. We first define

$$X = \#K22s(u_1v, u_2v) \quad \text{and} \quad X_o = \#openK22s(u_1v, u_2v).$$

We have

$$\mathbb{E}[X] = \sum_{forks} \#K22s(fork)\mathbb{P}(fork). \tag{3.13}$$

As each fork is chosen uniformly at random and as a K22 has two forks, we get

$$\mathbb{E}[X] = \sum_{forks} \#K22s(fork)\frac{1}{\#forks} = \frac{2\#K22s}{\#forks}. \tag{3.14}$$

Similarly,

$$\mathbb{E}[X_o] = \frac{\#openK22s}{\#forks}. \tag{3.15}$$

We may thus define two efficient unbiased estimates for #K22s and #openK22s:

$$Y = \frac{\#forks}{2n}\sum_{i=1}^{n} X_i. \quad \text{and} \quad Y_o = \frac{\#forks}{n}\sum_{i=1}^{n} X_{oi}. \tag{3.16}$$

We have $\mathbb{E}[Y] = \#K22s$ and $\mathbb{E}[Y_o] = \#openK22s$. The number of forks with a vertex v as a root is given by $\binom{d^-(v)}{2}$. The total number of forks in the TS is thus $\sum_{v\in V}\binom{d^-(v)}{2}$. Lastly, as we are interested in the

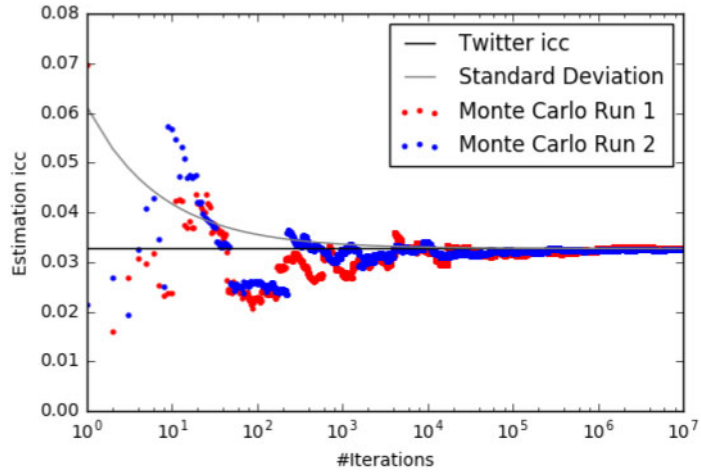

FIG. 4. Estimation of the clustering coefficient with Monte Carlo method.

interest clustering coefficient, we define

$$Z = \frac{4Y}{Y_0}. \tag{3.17}$$

As $\lim_{n\to\infty} Y = \#K22s$ and $\lim_{n\to\infty} Y = \#openK22s$, we have that $\lim_{n\to\infty} Z = icc$.

Experiments. We carried out two runs with 10 million iterations. It took about 2min30 for one run (60000 iterations per second). The value of the estimator of the icc for the two runs is plotted as a function of the number of iterations in Fig. 4. We first see that the estimator converges as expected to the value of the icc of TS represented by a straight horizontal line (and which was computed exactly in the previous section). We also plotted the estimated standard deviation as a function of the number of iterations. To obtain it, we did one billion iterations. We then estimated the standard deviation σ, and plotted $\frac{\sigma}{\sqrt{n}}$. We see that large jumps or discontinuity happen, but only at the beginning. They correspond to the draw of a fork with a lot of K22s and open K22s corresponding to a user who does not have the same icc as the global network. Then, the convergence is quick. After 300 iterations, the standard deviation is below 10% and after 1000 iterations, we do not experience a value of the runs less precise than 10%.

4. Results: clustering coefficients in Twitter

To compute the number of K22s and open K22s, directed triangles, and undirected triangles in the Twitter Snapshot, we used a cluster with a rack of 16 Dell C6420 dual-Xeon 2.20GHz (20 cores), with 192 GB RAM, all sharing an NFS Linux partition over Infiniband. It took 51 hours to compute the exact numbers of K22s and open K22s, corresponding to 265 h of cumulative computation times on the cluster. We reported the results in Table 1.

Number of K22s and triangles. We see that the numbers of K22s and open K22s are huge, 2.6×10^{16} and 3.1×10^{18}, respectively. It has to be compared with the number of triangles which are several orders of magnitude smaller: for example, 2.5×10^{12} and 1.3×10^{14} for transitive triangles.

TABLE 1 *Clustering coefficients (exact and approximated count) in the TS*

	# closed	# open	cc
icc	25,605,832,012,451,571	3,138,466,676,914,054,233	0.032634831
	2.6×10^{16}	3.1×10^{18}	3.3%
tcc	2,469,018,039,988	129,023,573,841,024	0.019136178
	2.5×10^{12}	1.3×10^{14}	1.9%
ccc	723,131,368,202	129,023,573,841,024	0.016813936
	7.2×10^{11}	1.3×10^{14}	1.7%
ucc	623,873,346,660	1,631,948,600,661,523	0.001146862
	6.23×10^{11}	$1{,}63 \times 10^{15}$	0.11%
mcc	317,649,850,664	8,924,125,201,234	0.106783526
	3.2×10^{11}	8.9×10^{12}	10.7%

TABLE 2 *Clustering coefficients without the mutual structures*

	icc	tcc	ccc	ucc
Twitter	3.1%	0.51%	0.24%	0.057%

Clustering coefficient in the mutual graph. The mutual graph captures the friendship relationships in the social network. The mutual clustering coefficient thus is high ($mcc = 10.7\%$), as cliques of friends are frequent in Twitter.

Clustering coefficients in the whole graph. We observe that $icc = 3.3\% > tcc = 1.9\% > ccc = 1.7\% > ucc = 0.11\%$. Directed metrics better capture the interest relationships in the TS as ucc is very low. The highest parameter is the icc. It confirms the hypothesis of this article that common interests between two users are better captured by the notion of K22 than by a direct link between these users. As expected, the second parameter is the one using transitive triangles. Indeed, they capture a natural way for a user of finding a new interesting user, that is, considering the followings of a following, especially after having seen retweets. A bit surprisingly, the ccc is not very low. In fact, a large fraction of the cyclic triangles are explained by corresponding triangles in the mutual graph (triangles of bi-directional links). A way to artificially take off the social influence in order to focus exclusively on the directed interest part of the graph is to remove the (open and closed) triangles and K22s contained in the mutual graph from the total count. Indeed, each undirected triangle of the mutual graph induces two cyclic triangles and four transitive triangles, and each undirected open triangle induces two open triangles. In the same way, each undirected K22 induces two K22s and each undirected open K22 induces two open K22s. The obtained results are shown in Table 2. If we take off those mutual triangles, both the tcc and the ccc values drop to 0.51% and 0.24%, respectively, while the icc stays about the same at 3.1%. This tends to confirm the hypothesis that the directed triangle clusterings somehow measure the friendship part of the TS more than the interest part.

We can even go one step further by computing the number of triangles in the graph in which all bidirectional edges have been removed. In that case, the ccc drastically drops to 0 (we found no cyclic triangles without at least a bidirectional arc in the dataset!). This confirms that cyclic triangles are created by friendship

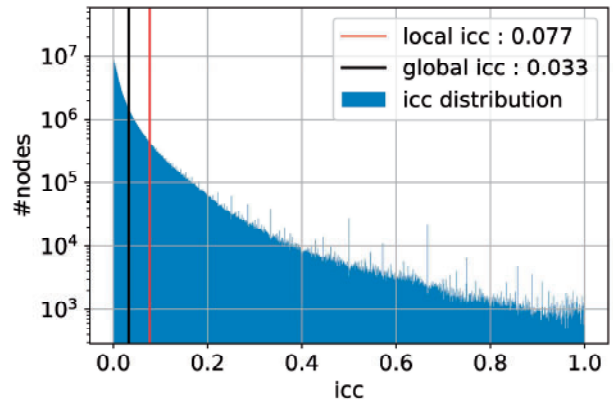

FIG. 5. Histogram of the distribution of the interest clustering coefficient over all users of the Twitter Snapshot. The vertical bars indicate the value of the glocal icc (3.3%) and the average value (7.7%) or local icc.

relations and that the ccc gives no information about the directed part of the graph. On the opposite icc and tcc increase to 4.2% and 3.6%, respectively, when considering only directed arcs. This confirms that icc and tcc provide important knowledge on the directed part.

Distribution of the icc and local clustering. We also provide the distribution of the values of the interest clustering coefficient over all users (having open K22s) in Fig. 5. We see that the icc greatly varies between 0 and 1. A large number of nodes have a low value of icc, for example, 2.23×10^7 users (10.2% of the users with open K22s) have a value of 0, meaning they are part of open K22s but not of K22s. At the opposite end, 2.4×10^4 users (0.011% of the users with open K22s) have a value of 1, meaning that all their open K22s belong to a K22. The *average value* is equal to 7.7%. This value could be used as a definition of a *local* $icc^{\top}$. Indeed, as discussed above, the number of K22s and open K22s per user have been computed while considering a user as a top vertex. A second local coefficient, $icc_{\perp}$, can be defined for bottom vertices.
Similarly to what was found in Facebook, the local coefficient has a larger value than the global one. This may be due to the fact that a large number of nodes with few K22s and open K22s (usually nodes with small degrees) only are in a single small strongly connected community, and thus have a higher than average icc. On the contrary, a small number of nodes with larger degrees and larger number of K22s and open K22s may be in different communities, leading to smaller than average icc.

5. Results: other directed datasets

We computed the different metrics on four other directed networks: two social networks, a web network and a citation network. The data information are gathered in Table 3, while the clustering coefficients are reported in Table 4. We also computed the values of the clustering coefficients without the mutual structures (not provided here); interestingly, those values are close to the ones on the total graphs.
We observe that the structure of each dataset is revealed by (the mix of) values of the different clustering coefficients, as discussed below.

Instagram: Instagram is a photo and video-sharing social network. This dataset was collected by Ferrara *et al.* [34] in 2014. The network is close to the Twitter one. Nodes corresponds to the accounts, and there is a link $u \to v$ if the account u follows the account v. The results are quite similar to what we found for

TABLE 3 *Datasets information. N is the number of nodes, $|E|$ the number of edges, and $\frac{|E|_m}{|E|}$ the fraction of edges implied in a bidirectional link.*

	Is a social network	N	$\|E\|$	$\frac{\|E\|_m}{\|E\|}$
Instagram	Yes	4.5×10^4	6.7×10^5	11%
Flickr	Yes	2.3×10^6	3.3×10^7	62%
Web (.edu)	No	6.9×10^5	7.6×10^6	25%
Citations	No	3.8×10^6	1.7×10^7	0%

TABLE 4 *Clustering coefficients of the directed datasets*

	icc	tcc	ccc	mcc	ucc
Instagram	12.0%	15.4%	3.7%	22.6%	4.1%
Flickr	12.4%	12.2%	9.3%	13.9%	10.8%
Web (.edu)	46.3%	59.6%	18.8%	78.5%	0.69%
Citations	22.3%	9.1%	0%	*(none)*	6.7%

Twitter: the *icc* and *tcc* are high and of the same order; the *ccc* is also high because of the bi-directional edges (it drastically drops to 0.06% when removing those links). The *mcc* is the highest value, while the *ucc* is lower than the others. This confirms that social networks share some common characteristics.

Flickr: Flickr is an image and video hosting service, which allows you to follow other people on the platform to see more easily their content. The dataset was collected in 2008 by Mislove *et al.* [35]. This is once again a graph of followers of a directed social network. The values are similar to the previous one but for the *ucc*, which is higher. We can notice that Flickr looks more like a social media than Twitter and Instagram, since there is 62% of links implied in bidirectional. This explains why the undirected clustering coefficient is not so different from the mutual one .

Berkley-Stanford.edu web pages: The dataset was collected in 2002 by Leskovec et al.[36]. The nodes represent the pages from berkely.edu and stanford.edu domains and directed edges represent hyperlinks between them. The *tcc*, *icc*, and *mcc* are really high. For the *tcc*, this is due to the very hierarchical structure of the institution web pages. As an example, a researcher will be linking towards his group, laboratory and university in its website, while the group website is linking to its laboratory and university... This strong structure translates into a high value of the *tcc*. As for the *icc*, research and educational domains form naturally strong communities creating large number of common neighbours for individuals of the same domain, and thus a high *icc*. Groups/teams/departments also constitutes strong social communities, leading to a high *mcc*.

Citations: Collected by Leskovec *et al.* [37], it includes all citations made by patents granted between 1975 and 1999. This is a good example of information network, giving a high value of icc of 22.6%, while the tcc value is 9.1%. Indeed, research fields and industry domains are strong communities leading

to a high icc. Moreover, it is also not rare to cite a patent and its citations (the patent acting as a survey), explaining the tcc value. Note that there are no cyclic triangles nor bidirectional links, because of the temporal structure of citations—a paper will only cite older papers.

Takeaways: The following takeaways summarize the variety of information given by the different clustering coefficients:

- A high value of icc indicates the presence of clusters of interests such as research communities or interest fields.
- A high value of tcc is the sign of an important *local* phenomena of friends' or acquaintances' recommendations and/or of a high hierarchical structure in the dataset.
- The ccc has no real social meaning. If its value can be high in a directed graph, this is only due to the presence of bidirectional arcs and triangles. The closure of a cyclic triangles is very rare in directed networks with no bi-directional edges, confirming the general intuition.
- Directed networks have a high *mcc*. Indeed, their bidirectional parts (mutual graph) have strong social communities, leading to a high clustering coefficient.
- The ucc is usually significantly lower, showing that the directed part of the network is better understood using directed clustering coefficients.
- Directed social networks have similar mixes of values of their undirected and directed clustering coefficients, however, with some notable differences, due to their diverse usages and information.

6. Model with addition of K22s

To model complex networks, a model with a high number of triangles was introduced in [38]. In this section, we introduce a new random graph model in which the *number of K22s is higher* than classical directed random graphs. The model is based on the one from Bollobás *et al.* [39] to which we add what we call a K22 event. A K22 event closes an open K22. The principle is that if a user u has a common interest with a user v, and if the user u has another interest, then v has an increased probability to be also interested and to follow it. We show that the in-degree and out-degree distributions of the introduced model follow a power law (as many real networks). Lastly, we show that the interest clustering coefficient of the generated graphs increase when we increase the probability of a K22 event.

6.1 *Presentation of the model*

We recall here the events defining the classic preferential attachment model of [39] and define the K22 event. We start with an initial graph $G_0 = (V_0, E_0)$. Then, at each time step t:

- With a probability (1-p) (**Bollobás *et al.* event**):
  - With a probability α, we add a node u and a link leaving this node and reaching an existing node v chosen with a probability proportional to $d_{in}(v) + \delta_{in}$;
  - With a probability β, we add a node v and a link reaching this node and leaving an existing node u chosen with a probability proportional to $d_{out}(u) + \delta_{out}$;
  - With a probability $1 - \alpha - \beta$, we add an edge between two existing nodes, chosen with probability proportional to $d_{out}(u) + \delta_{out}$ for the leaving node u and $d_{in}(v) + \delta_{in}$ for the reached node v.

- With a probability p (**K22 event**):
  1) We choose a random node (called u_1) with a probability proportional to its out-degree $d_{\text{out}}(u_1)$;
  2) We pick uniformly at random an out-neighbour of the node u_1 (called v_1);
  3) We pick uniformly at random an in-neighbour of the node v_1 (called u_2);
  4) We pick uniformly at random an out-neighbour of the node u_2 (called v_2);
  5) We add a link from u_1 to v_2.

The idea of the K22 event is to close an open K22; since u_2 follows v_1 and v_2 at the same time, v_1 and v_2 have a higher probability to be similar, and an account u_1 following v_1 has a higher chance to be interested in v_2.
Note that it is possible to introduce multiedges with the K22 events. Indeed, to make the problem tractable, we allow $u_1 = u_2$ in Step 3), or $v_2 = v_1$ in Step 4). In the empirical study, we construct the random graphs with the multiedges then get rid of them at the end of the construction. We empirically verified that the multiedges do not impact the results presented at the end of the section. Indeed, most of them appear for low degree nodes and, thus, they do not affect the tail of the degree distributions.

6.2 *In-degree and out-degree distributions*

We show in what follows that the in- and out-degree distributions of the introduced model follow power-laws, as most real networks. More precisely:

THEOREM 6.1 The probability $P(i)$ (respectively $P(o)$) for a node to have in-degree i (resp. out-degree o) in the new model is:

$$P(i) \underset{i>>1}{\sim} i^{-(1+\frac{1}{A})} \quad \text{and} \quad P(o) \underset{o>>1}{\sim} o^{-(1+\frac{1}{B})},$$

where $A = p + \frac{(1-p)(1-\beta)}{1+(1-p)(\alpha+\beta)\delta_{in}}$ and $B = p + \frac{(1-p)(1-\alpha)}{1+(1-p)(\alpha+\beta)\delta_{out}}$.

Proof. We first focus on the in-degree distribution. This result is derived from the equation giving the evolution of the number of nodes of in-degree i as a function of time, sometimes called Master Equation.

Let $G(t) = (V(t), E(t))$ be the graph obtained at time t, and $N(t) = |V(t)|$. The number of edges at time t is $|E(t)| = t + |E_0| \approx t$, while the number of nodes is $N(t) = (1-p)(\alpha+\beta)(t+|V_0|) \approx (1-p)(\alpha+\beta)t$ when t is high enough. Hence, the mean in-degree (and out-degree) of the network is $m = \frac{1}{(1-p)(\alpha+\beta)}$.

Let us compute the in-degree distribution. Calling $N(i,t)$ the number of nodes of in-degree i at time t, we can write the Master equation:

$$N(i,t+1) - N(i,t) = (1-p)\alpha\delta_{0,i} + (1-p)\beta\delta_{1,i} + (1-p)(1-\beta)\frac{i-1+\delta_{in}}{\sum_{i=0}^{+\infty} N(i,t)(i+\delta_{in})} N(i-1,t)$$

$$- (1-p)(1-\beta)\frac{i+\delta_{in}}{\sum_{i=0}^{+\infty} N(i,t)(i+\delta_{in})}N(i,t)$$
$$+ p\frac{i-1}{\sum_{i=0}^{+\infty} N(i,t)i}N(i-1,t) - p\frac{i}{\sum_{i=0}^{+\infty} N(i,t)i}N(i,t), \tag{6.1}$$

where $\delta_{i,j}$ is the Kronecker delta.
The Master equation formulates the variation of the number of nodes with degree i between time t and time $t+1$. The two first terms on the right-hand side correspond to the addition of a new node, with degree 0 or 1 (depending on if we are in the first or second case of the Bollobás *et al.* event). The third and fourth terms are the probabilities that, during the Bollobás *et al.* event, an edge is connected to a node of degree $(i-1)$ or i. This event leads to the arrival of a new node of degree i, or the loss of one of them. Those events occur with probability $(1-p)(\alpha+(1-\alpha-\beta))$. Finally, the last two terms correspond to the probability that an edge connects to a node of degree $(i-1)$ or i during the K22 event.
We now show that the probability to connect to a node v_2 of a given degree after following an open K22 is proportional to the degree of this node. Indeed, the probability to connect to a node v_2 of a given degree after following an open K22 is

$$P(x=v_2) = \sum_{y\in N^+(v_2)} P(y=u_2)\times\frac{1}{d_{\text{out}}(y)}, \tag{6.2}$$

where $N^+(v_2)$ is the set of in-neighbours of v_2, and u_2 is defined in the model. Using the same reasoning, we have

$$P(x=u_2) = \sum_{y\in N^-(u_2)} P(y=v_1)\times\frac{1}{d_{in}(y)} \tag{6.3}$$

and

$$P(x=v_1) = \sum_{y\in N^+(v_1)} P(y=u_1)\times\frac{1}{d_{out}(y)}. \tag{6.4}$$

Since $P(y=u_1)=\frac{d_{\text{out}}(y)}{t}$, we deduce that

$$P(x=v_2) = \frac{d_{in}(x)}{t}, \tag{6.5}$$

which gives us the expected result.
Using this property and knowing that

$$\sum_{i=0}^{+\infty} i\cdot N(i,t) = |E(t)| = t \tag{6.6}$$

and

$$\sum_{i=0}^{+\infty} N(i,t)\delta_{in} = \delta_{in}N(t) = (1-p)(\alpha+\beta)\delta_{in}, \tag{6.7}$$

we can rewrite Equation 6.1 as:

$$\begin{aligned} N(i,t+1) &= \alpha\delta_{0,i} + \beta\delta_{1,i} \\ &+ \Big(p\frac{i-1}{1} + (1-p)(1-\beta)\frac{i-1+\delta_{in}}{1+(1-p)(\alpha+\beta)\delta_{in}}\Big)\frac{N(i-1,t)}{t} \\ &- \Big(1 + \big(p\frac{i}{1} + (1-p)(1-\beta)\frac{i+\delta_{in}}{1+(1-p)(\alpha+\beta)\delta_{in}}\big)\frac{1}{t}\Big)N(i,t). \end{aligned} \tag{6.8}$$

Let us call

$$Z \equiv 1 + (1-p)(\alpha+\beta)\delta_{in}. \tag{6.9}$$

We need the following lemma from [40]:

LEMMA 6.1 ([40]) If we have an equation of the form:

$$N(i,t+1) = \Big(1 - \frac{b(t)}{t}\Big)N(i,t) + g(t), \tag{6.10}$$

where $b(t) \to b$ and $g(t) \to g$ as $t \to +\infty$, then

$$\frac{N(i,t)}{t} \to \frac{g}{b+1}. \tag{6.11}$$

Using Lemma 6.1 and calling $P(i) = \lim_{t\to+\infty} \frac{N(i,t)}{t}$, we have:

$$P(i) = \frac{\big(\frac{(1-p)(1-\beta)}{Z} + p\big)(i-1) + \frac{\delta_{in}}{Z}}{1 + \big(\frac{(1-p)(1-\beta)}{Z} + p\big)i + \frac{\delta_{in}}{Z}}P(i-1). \tag{6.12}$$

Let us call

$$A \equiv \frac{(1-p)(1-\beta)}{Z} + p. \tag{6.13}$$

We thus have:

$$P(i) \qquad = \frac{i - 1 + \frac{\delta_{in}}{ZA}}{i + \frac{\delta_{in}}{ZA} + \frac{1}{A}}P(i-1) \tag{6.14}$$

$$= P(1) \prod_{k=2}^{i} \frac{k - 1 + \frac{\delta_{in}}{ZA}}{k + \frac{\delta_{in}}{ZA} + \frac{1}{A}} \tag{6.15}$$

$$= \frac{\Gamma(i + \frac{\delta_{in}}{ZA})\Gamma(\frac{1}{A} + \frac{\delta_{in}}{ZA} + 2)}{\Gamma(i + \frac{\delta_{in}}{ZA} + \frac{1}{A} + 1)\Gamma(\frac{\delta_{in}}{ZA} + 1)}. \tag{6.16}$$

Leading to

$$P(i) \underset{i>>1}{\sim} i^{-(1+\frac{1}{A})}. \tag{6.17}$$

The *out-degree distribution* calculation follows the same method. The master equation is the same, except that δ_{in} and β are replaced by δ_{out} and α. The slope of the out-degree distribution is thus:

$$P_{\text{out}}(o) \underset{o>>1}{\sim} o^{-(1+\frac{1}{B})}, \text{with } B = \frac{(1-p)(1-\alpha)}{1 + (1-p)(\alpha+\beta)\delta_{\text{out}}} + p. \tag{6.18}$$

Concentration. We have studied here the mean of the distributions. We now use the Azuma's inequalities to show the concentration around the mean. We have the following result [40]: let X_t be a martingale with $|X_s - X_{s-1}| \leq c$ for $1 \leq s \leq t$. Then:

$$P(|X_t - X_0| > x) \leq \exp(-x^2/2c^2t). \tag{6.19}$$

Let $Z(i,t)$ be the number of vertices of degree i at time t and let F_s denote the σ-field generated by the choices up to time s. We apply the result to $X_s = E(Z(i,t)|F_s)$. We have that $|X_s - X_{s-1}| \leq 2$. Indeed, when we add an edge in the network, we affect only the degrees of its two end-vertices. Since $Z(i,0) = E(Z(i,t))$, using the result with $x = \sqrt{t\log(t)}$, we have

$$P(|Z(i,t) - E(Z(i,t))| > \sqrt{t\log(t)}) \leq t^{-\frac{1}{8}}. \tag{6.20}$$

And hence, $\frac{Z(i,t)}{t} \underset{t\to+\infty}{\to} P(i)$ in probability. □

The degree distributions of the model follow power-laws, with exponents between -2 and $-\infty$. We notice that, for $p = 0$, we recover the exponents of the Bollobás *et al.* model $-(1 + \frac{1+(\alpha+\beta)\delta_{in}}{1-\beta})$ and $-(1 + \frac{1+(\alpha+\beta)\delta_{out}}{1-\alpha})$ [39], while, when p goes to 1, the exponent goes to -2.

Note that, similarly to the Bollobás *et al.* model, we cannot generate graphs with any wanted mean-degree and fixed slopes of the power-law. Some constraints exist in order to keep $\delta_{\text{in}} > 0$ and $\delta_{\text{out}} > 0$. For instance, with $\alpha = \beta = 0.4$ and slopes of -2.5 (the values of our experiments), p has to stay in the interval $[\frac{1}{6}, \frac{2}{3}]$.

Validation by simulations. We validate the analysis and the hypothesis by simulation. In Figure 6, we present the in- and out-degree distributions of a network built with our new model as an example. The parameters are fixed to $p = 0.5$, $\alpha = \beta = 0.4$, and $\delta_{\text{in}} = \delta_{\text{out}} = 2.0$. In this case, the expected slopes are -2.5. The fit is almost perfect: -2.509 and -2.498 for the in- and out-degree distributions.

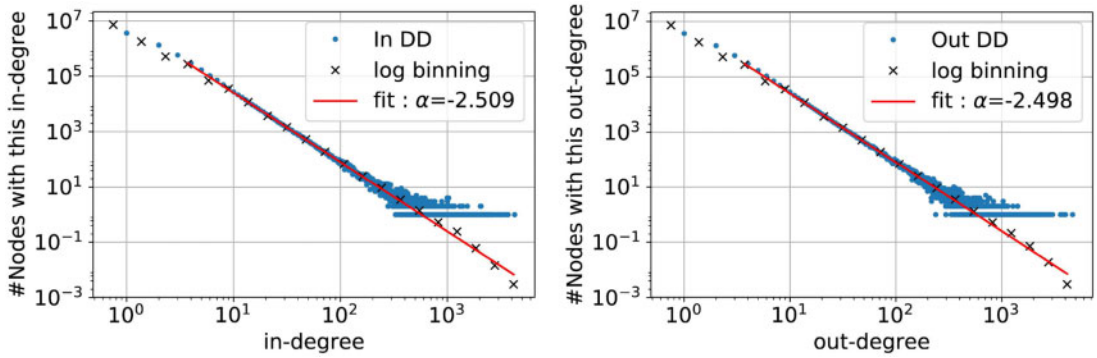

FIG. 6. In- (Left) and out- (Right) degree distributions of a network built with the new model. The obtained distribution is given by the blue points; the black crosses represent the logarithmic binning of the distribution (a mean of a given amount of points on a logarithmic scale). The red straight line is the fit of the logarithmic binning; it has slope of -2.509 (respectively -2.498) for the in- (respectively out-) degree distribution (expected slopes from analysis are -2.5).

6.3 *Interest clustering coefficient of the new model*

We show by simulation how the icc increases as p increases. We compare it with the one of the Bollobás et al. model. Note that, when p increases, the average degree of the model increases. Indeed, the mean degree of the new model is $m_{\text{new}} = \frac{1}{(1-p)(\alpha+\beta)}$. To compare networks with the same characteristics (mean degrees and exponents of the in-degree distribution), we adapt the parameters of the second model with the value of p.

////Since, in the Bollobas *et al.* model, the mean degree is $m_{Bol} = \frac{1}{\alpha+\beta}$, we can compare the two models by choosing values α_{new}, β_{new}, and p for our model. This imposes a value of m_{new}. We then choose α_{Bol}, β_{Bol} for the Bollobás et al. model, so that the two networks have the same mean degree. Finally, we choose δ_{in} so that the exponent of the in-degree distribution stays the same in both networks. In practice, we have fixed the exponent to -2.5 and imposed $\alpha_{new} = \beta_{new} = 0.4$. We compare the icc for both models for different values of p and report the results in Figure 7. We used graphs of size $N = 10^7$ nodes and averaged over 10 networks for each point. We see that the icc varies from 0.036% to 4.4% when p varies from 0.2 to 0.6.

Let us notice that imposing the value of the slope results in some constraints for the values of the other parameters. Indeed, solving the slope equations from Theorem 6.1 with $\alpha, \beta, \delta_{in}$, and $\delta_{out} > 0$ imposes a maximum value for p. For instance, with a slope of -2.5, the maximum value for p is 0.65. Additionally, the value of the icc is at most 0.045 for such a slope value. This is high enough to allow us to obtain an adequate model of Twitter. However, it may be not sufficient to model some other datasets with higher icc values. It would be an interesting future work to propose other attachment models leading to higher values of icc.

7. Link recommendation

We propose to use the K22s defined for our metric to carry out link recommendation, as we advocate that the interest clustering coefficient is a good measure of common user interests. For a neighbour, the principle is to recommend links closing open K22s. We define the *strength of a link* as the number of open K22s it would close if added to the graph. Links are then recommended by decreasing strengths.

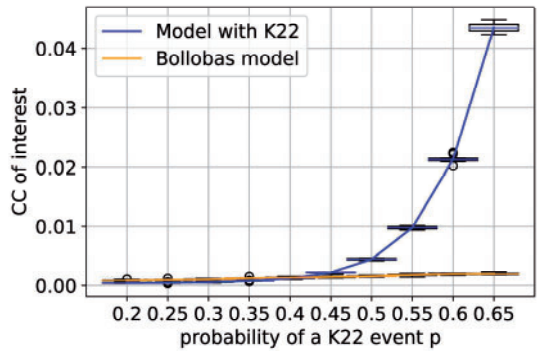

FIG. 7. Interest clustering coefficient of our new model as a function of p, the probability of a K22 event. The value is compared with the one of the Bollobás *et al.* model [39].

Typical recommendation systems propose the strongest link to a user (e.g. Facebook) or a top 10/top 20 list (e.g. Youtube).

We tested our method on the Twitter snapshot. We considered a population of 1000 users selected uniformly at random over the full population of Twitter's users. Note that we excluded users following no one. Indeed, isolated users are not interesting users per se and for this study and they have no TT or K22 recommendations.

For each node, we computed its open K22s: for a node x, we follow all its out-neighbours, then for each out-neighbour, we follow its in-neighbours, then for each in-neighbour, we follow its out-neighbours. These last nodes (which were not already followed by x) are the recommended nodes. We then count how many times a node is recommended. This gives the link strength.

We compared the method with classic recommendations using triangles. For example, on Facebook, it is frequent to have a message such as '8 of your friends know Bob. Do you know Bob?' Connecting with Bob would close 8 open (undirected) triangles. As we are considering a directed graph and are focusing on interest links, we computed recommendations based on transitive triangles, as they have more social sense than cyclic triangles. For a user x, we recommend the out-neighbours of the out-neighbours of x.

Note that there are a lot more open K22s than open triangles in the graph, 3.1×10^{18} compared to 1.3×10^{14}. We argue in the following that it allows to make more recommendations and, most importantly, better recommendations.

We report in Fig. 8 histograms of the cumulative distribution over the 1000 random users of the strengths of the recommendation with maximum strength and of the 10th recommendation. The top plots present K22 recommendations while the bottom ones the TT recommendations. The right plots show the complete cumulative distribution in log scale, while the left plots are a zoom on recommendations with weak strengths (≤ 20). Beware that the y-scale of the K22 zoom left plot which is between 0 and 0.1. Notice also the difference in x-scale for the right plots.

Top/Max recommendation. We remark that a small amount of users have TT recommendations and no K22 recommendation. This is due to the fact that for a user with few outgoing links, it is more probable that the followed users are also following at least one other user (providing a TT recommendation) than they are followed by other users (necessary to provide a K22 recommendation). We do not advocate to use only K22 recommendations, but to use it as a complementary tool. In particular, for users with no

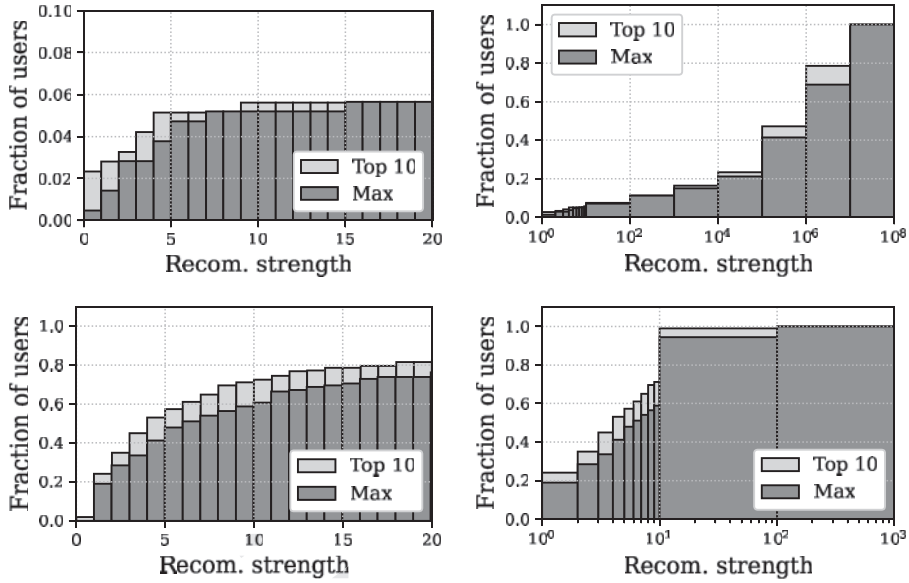

FIG. 8. Cumulative distribution of the max and 10th recommendation strength over 1000 random Twitter's users for K22 recommendation (Top) and transitive triangle recommendation (Bottom). The left plots are a zoom on recommendations with weak strengths (≤ 20). The right plots present the complete cumulative distribution in log scale. Beware of the y-scale for the K22 zoom left plot.

TT and K22, recommendations would only be made based on global social network statistics (trending topics for example).

However, when a K22 recommendation exists for a user, it has much more strength than the TT recommendations for her. Indeed, 21% of users have TT recommendations of strengths 0 or 1. This number is just 1.2% for K22 recommendations. A recommendation of strength 1 has very good chance to be of no interest, as it is based on the following of a single user over 500 million ones. Similarly, 28% of users only have TT recommendations of strength 2 or lower (to be compared with 2.5% for K22 recommendations). This means that, for a very large portion of users, TT recommendations are based on very few links. On the contrary, more than 94% of users have a top K22 recommendation with strength more than 10. *We are thus able to carry out a meaningful recommendation for the vast majority of users using K22s.*

Top 10 recommendations. When considering a recommendation system proposing a top 10, we see that 25% of users have their 10th TT-recommendation of strength 1 or lower, and 35% of strength 2 or lower. There does not exist a significant top 10 list for more than one third of users. On the contrary, 94% of users have their 10th K22-recommendation with strength higher than 10. Top 10 recommendation systems can thus be implemented for most users using K22s. Moreover, the distribution of recommendation strengths is very flat when using TT (a large number of top recommendations have strength 1), see Fig. 9. Thus, it is very hard to discriminate between recommended users and to do a meaningful ranking of recommendations. At the opposite end, the distribution usually is steep for K22. It is thus a lot easier to establish a ranking.

Typical users. We present in Fig. 9 the strengths of the top 10 recommendations using K22 (Left) and TT (Right) for four typical users. For the first one (Top), it is implicated in around 200 triangles, representing

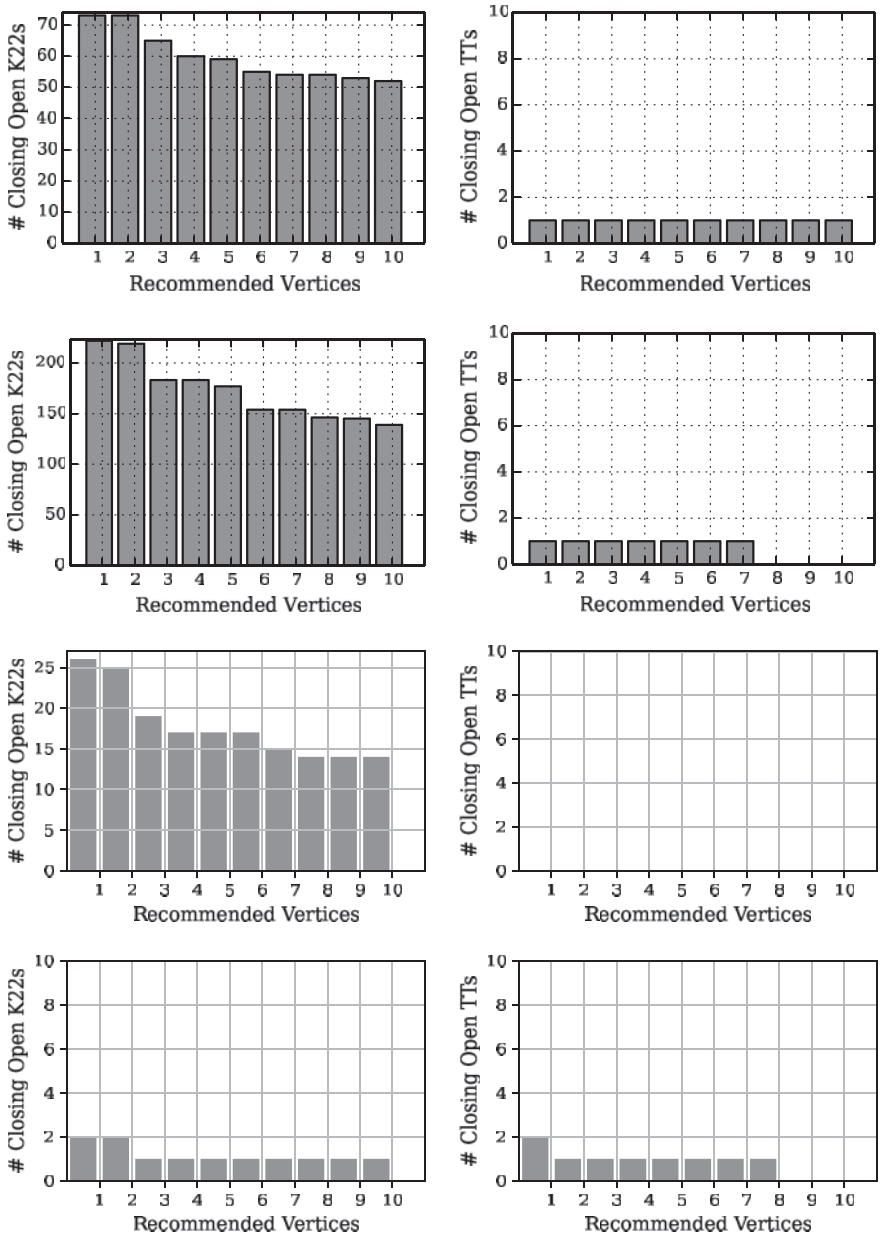

FIG. 9. Strengths of the top 10 recommendations for 4 typical Twitter users using (left) K22 recommendations (right) TT recommendations.

each a potential recommendation. However, the strength of the recommendations is very low, just 1 for all of them. Recommendations for this user would be very bad for two reasons: first, they are based on the choice of only 1 user. Second, if the recommendation system had to propose a top 10, how would

it discriminate between the 200 similar potential ones with similar strength. On the contrary, the K22 recommendations have much more strengths: 72 for the 1st and the 2nd ones, and 52 for the 10th one. The K22 recommendations are thus much more well-grounded. For the second user (Middle Top), we observe a similar phenomenon, but with fewer recommendations. It is not even possible to build a top 10 for her using TT as only 8 links can be proposed, and not with a high confidence (strength 1). Conversely, the top 10 K22 recommendations have strengths between 215 and 135. The third user (Middle Bottom) does not have any open TT, thus no recommendation based on triangles would be possible. In contrary there is enough K22s to present a satisfying top 10 K22 recommendations—even the 10th recommendation is still based on 14 open K22s. Finally, the fourth user (Bottom) does not have enough open TTs nor open K22s to do solid recommendations. Still, the number of open K22s is high enough to have 10 (weak) recommendations, while we can only recommend 8 people with the TT recommendations.

8. Conclusion

The clustering coefficient metric has been widely studied in real-world social networks in order to exhibit the presence of interactions between friends. Indeed, the classical clustering coefficient apprehends the social phenomena that my friends tend to be connected with each others. However, it is not adequate to take into account directed interest links. This leads us to introduce a new metric, the *interest clustering coefficient*, to capture the interest phenomena in a directed graph. The interest clustering coefficient is based on the idea that, if two people are following a common neighbour, they have a higher chance to share other common neighbours, since they have at least one interest in common.

We computed this new metric on a snapshot of Twitter from 2012 with 505 million users and 23 billion links, known to be both a social and information media. The computation was made on the total graph, giving the exact value of the interest clustering coefficient, and using two sampling methods: one using edge-sampling graphs, and one using a Monte-Carlo algorithm. We also computed undirected and directed clustering coefficients introduced in the literature and based on triangles, namely the *mutual clustering coefficient* and the *undirected clustering coefficient* (undirected ones) [41], as well as the *transitive clustering coefficient* and the *cyclic clustering coefficient* (directed ones) [6]. We also computed those various coefficients on the Twitter graph after having removed bidirectional edges, in order to separate the interest and the social part of the network. While doing so, we observed that the interest clustering coefficient almost stayed the same, while the ones based on triangles significantly decreased. Those results on the Twitter graph consolidate the idea that Twitter is indeed used as a social and information media, and that the interest clustering coefficient is well-suited to capture the interest part of directed networks. We also computed the interest clustering coefficient, the undirected and the directed clustering coefficients introduced in the literature on different directed datasets, either social networks or information networks.

The following takeaways summarize our conclusions:

- A high value of interest clustering coefficient indicates the presence of clusters of interests in the studied graph;
- A high value of transitive clustering coefficient indicates the presence of social clusters in the graph, but this clustering might be increased due to a friends' or acquaintances' recommendations system, as found for instance in Twitter;
- A high value of cyclic clustering coefficient might be due only to the presence of bidirectional arcs and triangles—the metric thus is not well-suited for directed graph;

- Bidirectional edges mainly represent the social part of directed networks. Thus, the subgraph containing only bidirectional links (called *mutual graph*) has strong social communities, leading to a high mutual clustering coefficient;
- The undirected clustering coefficient usually has significantly lower values—this metric built for undirected networks is not able to capture the information of directed ones.

In the article, we finally proposed a new model, able to build random directed networks with a high value of K22s, as well as a new method for link recommendation using K22s.

We have only begun the study on this last point. As a future work, we would like to investigate it further, as we find it really promising. Indeed, while link recommendation is a deeply studied topic, a large number of its applications in real-life systems are based on closing triangles. For instance, Twitter uses a 'people you follow also follows...' recommendation system, thus based on closing transitive triangles. The recommendation method presented in Section 7 brings another way to discover missing links, based on the share of common interests. Moreover, the huge number of open K22s—over 200 times more than open triangles—drastically increases the accuracy of this method compared to the triangle one. In particular, it would be interesting to carry out a real-world user case study to investigate if users are more satisfied by such recommendations.

Funding

This work was supported by the French government through the UCA JEDI [ANR-15-IDEX-01], EUR DS4H [ANR-17-EURE-004] Investments in the Future projects, and ANR DIGRAPHS, by the SNIF project, and by Inria associated team EfDyNet. The authors are grateful to the OPAL infrastructure from UniversitÃ© CÃ´te d'Azur for providing resources and support.

References

1. Shaban, H. https://www.washingtonpost.com/technology/2019/02/07/twitter-reveals-its-daily-active-user-numbers-first-time/.
2. Gabielkov, M., Rao, A. & Legout, A. (2014) Studying social networks at scale: macroscopic anatomy of the twitter social graph. *ACM SIGMETRICS Performance Evaluation Review*, vol. 42. ACM, pp. 277–288.
3. Myers, S. A., Sharma, A., Gupta, P. & Lin, J. (2014) Information network or social network?: The structure of the twitter follow graph. *Proceedings of the 23rd International Conference on World Wide Web*. ACM, pp. 493–498.
4. Mislove, A., Marcon, M., Gummadi, K. P., Druschel, P. & Bhattacharjee, B. (2007) Measurement and analysis of online social networks. *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement*. ACM, pp. 29–42.
5. Ugander, J., Karrer, B., Backstrom, L. & Marlow, C. (2011) The anatomy of the Facebook social graph. *arXiv preprint arXiv:1111.4503*.
6. Fagiolo, G. (2007) Clustering in complex directed networks. *Phys. Rev. E*, **76**, 026107.
7. Kwak, H., Lee, C., Park, H. & Moon, S. (2010) What is Twitter, a social network or a news media? *Proceedings of the 19th International Conference on World Wide Web*. ACM, pp. 591–600.
8. Strogatz, S. H. (2001) Exploring complex networks. *Nature*, **410**, 268.
9. Lü, L. & Zhou, T. (2011) Link prediction in complex networks: a survey. *Physica A*, **390**, 1150–1170.
10. Albert, R. & Barabási, A.-L. (2002) Statistical mechanics of complex networks. *Rev. Modern Phys.*, **74**, 47.
11. Backstrom, L., Boldi, P., Rosa, M., Ugander, J. & Vigna, S. (2012) Four degrees of separation. *Proceedings of the 4th Annual ACM Web Science Conference*. ACM, pp. 33–42.

12. Leskovec, J. & Horvitz, E. (2008) Planetary-scale views on a large instant-messaging network. *Proceedings of the 17th International Conference on World Wide Web*. ACM, pp. 915–924.

13. Watts, D. J. & Strogatz, S. H. (1998) Collective dynamics of âĽ˜small-worldâĽ™ networks. *Nature*, **393**, 440.

14. Clauset, A., Shalizi, C. R. & Newman, M. E. (2009) Power-law distributions in empirical data. *SIAM Rev.*, **51**, 661–703.

15. Boykin, P. O. & Roychowdhury, V. P. (2005) Leveraging social networks to fight spam. *Computer*, **38**, 61–68.

16. Chen, J., Geyer, W., Dugan, C., Muller, M. & Guy, I. (2009) Make new friends, but keep the old: recommending people on social networking sites. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, pp. 201–210.

17. Silva, N. B., Tsang, R., Cavalcanti, G. D. & Tsang, J. (2010) A graph-based friend recommendation system using genetic algorithm. *2010 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, pp. 1–7.

18. Granovetter, M. S. (1977) The strength of weak ties. *Social Networks*. Elsevier, pp. 347–367.

19. Newman, M. E. (2003) Ego-centered networks and the ripple effect. *Soc. Netw.*, **25**, 83–95.

20. Kim, B. J. (2004) Performance of networks of artificial neurons: the role of clustering. *Phys. Rev. E*, **69**, 045101.

21. Barrat, A. & Weigt, M. (2000) On the properties of small-world network models. *Eur. Phys. J. B*, **13**, 547–560.

22. Opsahl, T. & Panzarasa, P. (2009) Clustering in weighted networks. *Soc. Netw.*, **31**, 155–163.

23. Saramäki, J., Kivelä, M., Onnela, J.-P., Kaski, K. & Kertesz, J. (2007) Generalizations of the clustering coefficient to weighted complex networks. *Phys. Rev. E*, **75**, 027105.

24. Rossi, R. & Ahmed, N. (2015) The network data repository with interactive graph analytics and visualization. *AAAI*, **15**, 4292–4293.

25. Alon, N., Yuster, R. & Zwick, U. (1997) Finding and counting given length cycles. *Algorithmica*, **17**, 209–223.

26. Coppersmith, D. & Winograd, S. (1987) Matrix multiplication via arithmetic progressions. *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*. ACM, pp. 1–6.

27. Le Gall, F. (2014) Powers of tensors and fast matrix multiplication. *Proceedings of the 39th International Symposium on Symbolic and Algebraic Computation*. ACM, pp. 296–303.

28. Latapy, M. (2008) Main-memory triangle computations for very large (sparse (power-law)) graphs. *Theoret. Comput. Sci.*, **407**, 458–473.

29. Schank, T. & Wagner, D. (2005) Finding, counting and listing all triangles in large graphs, an experimental study. *International Workshop on Experimental and Efficient Algorithms*. Springer, pp. 606–609.

30. Kolountzakis, M. N., Miller, G. L., Peng, R. & Tsourakakis, C. E. (2012) Efficient triangle counting in large graphs via degree-based vertex partitioning. *Internet Math.*, **8**, 161–185.

31. Wang, J., Fu, A. W.-C. & Cheng, J. (2014) Rectangle counting in large bipartite graphs. *2014 IEEE International Congress on Big Data*. IEEE, pp. 17–24.

32. Sanei-Mehri, S.-V., Sariyuce, A. E. & Tirthapura, S. (2018) Butterfly counting in bipartite networks. *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, pp. 2150–2159.

33. Gabielkov, M. & Legout, A. (2012) The complete picture of the Twitter social graph. *Proceedings of the 2012 ACM Conference on CoNEXT Student Workshop*. ACM. pp. 19–20.

34. Ferrara, E., Interdonato, R. & Tagarelli, A. (2014) Online popularity and topical interests through the lens of instagram. *Proceedings of the 25th ACM Conference on Hypertext and Social Media*. ACM, pp. 24–34.

35. Mislove, A., Koppula, H. S., Gummadi, K. P., Druschel, P. & Bhattacharjee, B. (2008) Growth of the Flickr social network. *Proceedings of the 1st ACM SIGCOMM Workshop on Social Networks (WOSN'08)*.

36. Leskovec, J., Lang, K. J., Dasgupta, A. & Mahoney, M. W. (2009) Community structure in large networks: natural cluster sizes and the absence of large well-defined clusters. *Internet Math.*, **6**, 29–123.

37. Leskovec, J., Kleinberg, J. & Faloutsos, C. (2005) Graphs over time: densification laws, shrinking diameters and possible explanations. *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*. ACM, pp. 177–187.

38. Vázquez, A. (2003) Growing network with local rules: preferential attachment, clustering hierarchy, and degree correlations. *Phys. Rev. E*, **67**, 056104.

39. BOLLOBÁS, B., BORGS, C., CHAYES, J. & RIORDAN, O. (2003) Directed scale-free graphs. *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms.* Society for Industrial and Applied Mathematics, pp. 132–139.
40. DURRETT, R. (2007) *Random Graph Dynamics*, vol. 200. Cambridge: Cambridge University Press.
41. NEWMAN, M. E. (2003) The structure and function of complex networks. *SIAM Rev.*, **45**, 167–256.