

IMT School for Advanced Studies Lucca

Lucca, Italy

**Essays on statistical methods for the analysis of
social networks**

Ph.D. Program in Institution, Markets and Technologies
Curriculum in Economics, Management and Data Science

XXXI Cycle

By

Carolina Becatti

2019

The dissertation of Carolina Becatti is approved.

Program Coordinator:

Prof. Massimo Riccaboni, IMT School for Advanced Studies Lucca

Supervisors:

Prof. Guido Caldarelli, IMT School for Advanced Studies Lucca

Prof. Irene Crimaldi, IMT School for Advanced Studies Lucca

Dr. Fabio Saracco, IMT School for Advanced Studies Lucca

The dissertation of Carolina Becatti has been reviewed by:

Prof. Monia Lupparelli, University of Florence

Prof. Yamir Moreno, University of Zaragoza

IMT School for Advanced Studies Lucca

2019

Ai miei nonni

Contents

List of Figures	xi
List of Tables	xiv
Acknowledgements	xvi
Vita and Publications	xxiii
Abstract	i
1 Introduction to Complex Networks	1
1.1 The basics	2
1.1.1 Adjacency matrix	4
1.1.2 Bipartite networks	7
1.1.3 Nodes' degrees	9
1.1.4 Network quantities	10
1.2 Network models	13
1.2.1 Erdős-Rényi Random Graph	14
1.2.2 Barabási-Albert model	17
1.2.3 Fitness model	22
1.2.4 Randomizing a real-world network	23
1.3 The role of null models	31
1.3.1 Quantifying nodes' similarity in bipartite networks	33
1.4 Outline of the work	37

2	Entropy-based randomization of rating networks	39
2.1	Introduction	41
2.2	The model	43
2.3	Discussion on the model	47
2.3.1	Computational complexity	48
2.3.2	Possible extensions	50
2.4	General methodology for the analysis	50
2.4.1	Higher order topological benchmark	51
2.4.2	Monopartite projections	53
2.5	Datasets	55
2.6	Results	56
2.6.1	Monopartite Communities	58
2.7	Discussion and conclusions	69
3	Extracting significant signals of news consumption from social networks: the case of Twitter in the Italian political elections	72
3.1	Introduction	74
3.2	Dataset	77
3.3	General methodology for the analysis	77
3.3.1	Classification of verified users	78
3.3.2	Classification of non-verified users	81
3.3.3	Influence analysis	82
3.4	Results	89
3.4.1	Groups of verified accounts	89
3.4.2	Polarization of non-verified accounts	91
3.4.3	Influence analysis	92
3.5	Discussion and conclusions	95
4	Collaboration and followership: a stochastic model for activities in social networks	111
4.1	Introduction	113
4.2	The model	116
4.3	Discussion on the model	119
4.3.1	The model parameters	119
4.3.2	Meaning of the random weights	120

4.4	General methodology for the analysis	123
4.4.1	Estimation of the model parameters	123
4.4.2	Check of the asymptotic behaviors	124
4.4.3	Comparison between real and simulated network and relevance of the weights	124
4.4.4	Predictive power of the model	126
4.5	Datasets	128
4.6	Results	129
4.6.1	IEEE dataset for Automatic Driving	129
4.6.2	ArXiv dataset for Theoretical High Energy Physics	135
4.6.3	Instagram dataset	140
4.6.4	Summary of the results	145
4.7	Discussion and conclusions	146
5	Conclusions	149
A	Appendix	154
A.1	Bipartite Score Configuration Model	154
A.2	Bipartite Weighted Configuration Model with truncated sequence of weights	156
A.3	Bipartite Partial Score Configuration Model	158
A.4	Bipartite Score Random Graph	160
A.5	Further analyses of higher topological quantities: positive reviews only	162
B	Appendix	167
B.1	Bipartite Directed Configuration Model	167
B.2	Additional Tables and Figures	171
C	Appendix	177
C.1	Asymptotic behaviour of the total number of features	177
C.2	Asymptotic behaviour of the mean number of edges	178
C.3	Estimation of the model parameters	182
C.4	Data cleaning procedure	185
	References	187

List of Figures

1	Graphical representation of simple examples of undirected, directed and bipartite graphs	5
2	Examples of Poisson degree distributions with different average degrees	17
4	A toy example of the preferential attachment mechanism .	20
5	Graphical representation the V-motifs.	34
6	Graphical representation of a rating network.	42
7	Graphical comparison of the two definitions of probabilities: BiSCM vs Chung-Lu probabilities.	49
8	Checkerboard-like motifs.	53
9	Application of the method to the ML network: Checkerboard motifs	59
10	Application of the method to the ML network: Average-nearest-neighbour degree (I)	60
11	Application of the method to the ML network: Average-nearest-neighbour degree (II)	61
12	Application of the method to the ML network: Average-nearest-neighbour degree (III)	62
13	Application of the method to the ML network: Average-nearest-neighbour degree (IV)	63
14	Validated communities in the MovieLens dataset.	65
15	Validated communities in the Digital Music dataset.	67
16	Validated communities in the Anvur dataset.	69

17	Number of tweets and retweets per user per day	99
18	Graphical representation of an example of bipartite and directed network of tweets and retweets	100
19	Most used hashtags in the coalitions of M5S and informa- tion channels	101
20	Most used hashtags in the left-leaning and right-leaning coalitions	102
21	Distribution of the interactions towards the communities and of the polarization index values	103
22	Block structure of the biadjacency matrix	104
23	Validated and directed network of retweets.	105
24	Most retweeted users in group \mathcal{C}_1	106
25	Most retweeted users in group \mathcal{C}_2	107
26	Most retweeted users in group \mathcal{C}_3	108
27	Most retweeted users in group \mathcal{C}_4	109
28	Users' polarization after validation.	110
29	IEEE Automatic Driving: actions-features matrix	130
30	IEEE Automatic Driving: asymptotic behavior of the num- ber of features and edges in the actions-features network .	131
31	arXiv High Energy Physics: actions-features matrix	136
32	arXiv High Energy Physics: asymptotic behavior of the number of features and edges in the actions-features net- work	138
33	Instagram: actions-features matrix	141
34	Instagram: asymptotic behaviour of the number of fea- tures and edges in the actions-features network	142
35	Application of the method to the MI network: positive re- views only	164
36	Application of the method to the SM network: positive reviews only	165
37	Application of the method to the DM network: positive reviews only	166

38	Distribution of the polarization indices, conditioned on the total number of interactions observed for each users . .	171
39	Evolution of the mean number of edges along time (I) . . .	183
40	Evolution of the mean number of edges along time (II) . .	184

List of Tables

1	Data description	56
2	Some descriptive quantities in the network of retweets between verified and non-verified users.	80
3	IEEE Automatic Driving: estimation of the model parameters	130
4	IEEE Automatic Driving: comparison between real and simulated networks (I)	133
5	IEEE Automatic Driving: comparison between real and simulated networks (II)	133
6	IEEE Automatic Driving: predictions on the actions-features matrix	134
7	arXiv High Energy Physics: Estimates of the model parameters	136
8	arXiv High Energy Physics: comparison between real and simulated networks (I)	137
9	arXiv High Energy Physics: comparison between real and simulated networks (II)	137
10	arXiv High Energy Physics: predictions on the actions-features matrix	140
11	Instagram: estimation of the model parameters	143
12	Instagram: comparison between real and simulated networks (I)	143

13	Instagram: comparison between real and simulated networks (II)	144
14	Instagram: predictions on the actions-features matrix . . .	145
15	List of the most central users in the validated network of verified users	172
16	List of the most retweeted users within the subgraphs of M5S and information channels communities	173
17	List of the most retweeted users within the subgraphs of left- and right-leaning communities	174
18	List of the most retweeting users within the subgraphs of M5S and information channels communities	175
19	List of the most retweeting users within the subgraphs of left- and right-leaning communities	176

Material

This work is a collection of the following contributions:

- Chapter 1 is a review of definitions and methods regarding network theory, therefore each section is based on results already presented in articles and books belonging to the related literature. The sources and materials the text is referred to are reported in each paragraph.
- Parts of Chapter 1 overlap with [133, 134, 135] since this thesis is based on results presented in these articles.
- Chapter 2 and Appendix A reproduce [133]. The paper has been published on *Physical Review E*.
- Chapter 3 and Appendix B reproduce [134]. The paper is under review to *Palgrave Communications*.
- Chapter 4 and Appendix C reproduce [135]. The paper is under review to *Plos One*.
- Figures and Tables are a reproduction of the ones presented in [133, 134, 135].

Acknowledgements

I am grateful to my advisors *Guido Caldarelli*, *Irene Crimaldi*, *Fabio Saracco*, whose contribution was essential for writing this thesis. A sincere thanks also to *Monia Luppearelli* and *Yamir Moreno* for reviewing it, your precious advice and comments improved the quality of this work.

A special thought goes to *Irene*, working with her taught me a lot, and to *Fabio*, for always believing in my abilities and for the many opportunities he provided me during these years. Without him this work would definitely not have been possible, so thank you.

Thank you to my parents and friends for always being by my side. Their patience and support during good and bad times of the last three years mean the world to me. And finally, thank you to *Giuseppe* for always being my leading light.

Vita

August 7, 1991	Born, Bagno a Ripoli (Fi), Italy
Nov 2010 – Oct 2013	B.Sc. in Mathematics University of Siena, Italy
Nov 2013 – Oct 2015	M.Sc. in Statistics University of Siena, Italy
Feb – Jun, 2015	Erasmus+ Exchange Faculty of Economics and Business Katholieke Universiteit Leuven, Belgium
Nov 2015 – Jul 2019	Ph.D. in Economics, Management and Data Science IMT School for Advanced Studies Lucca, Italy
Mar – Jul, 2017	Teaching Assistant in Statistics Department of Economics and Management University of Pisa, Italy
Apr – Jun, 2018	Visiting Postgraduate Student Mathematical Institute University of Oxford, UK

Publications

1. C. Becatti, I. Crimaldi, F. Saracco, 2019. "Keywords dynamics in online social networks: a case-study from Twitter". Forthcoming in *Proceedings of the conference of the Italian Statistical Society " Smart statistics for smart applications"* (SIS 2019).
2. C. Becatti, G. Caldarelli, F. Saracco, 2019. "Entropy-based randomization of rating networks". *Physical Review E*, **99** 022306. [Link](#).
3. C. Becatti, G. Caldarelli, R. Lambiotte, F. Saracco, 2019. "Extracting significant signal of news consumption from social networks: the case of Twitter in Italian political elections". Under review to *Palgrave Communications*. arXiv preprint [arXiv:1901.07933](#).
4. C. Becatti, I. Crimaldi, F. Saracco, 2018. "Collaboration and followership: a stochastic model for activities in social networks". Under review to *Plos One*. arXiv preprint [arXiv:1811.00418](#).

Presentations

1. C. Becatti. “Keywords dynamics in online social networks: a case-study from Twitter”. Talk at the annual *Conference of the Italian Statistical Society “Smart statistics for smart applications”*, 19 – 21 June 2019, Milan (Italy).
2. C. Becatti. “Entropy-based randomization of rating networks”. Poster presentation at *IMT Research Symposium*, 10 – 11 July 2018, Lucca (Italy).
3. C. Becatti. “Entropy-based randomization of rating networks”. Poster presentation at *International School and Conference on Network Science*, 11 – 15 June 2018, Paris (France).
4. C. Becatti. “Spambot detection and polarization analysis: evidence from the Italian election Twitter data”. Seminar at Mathematical Institute, University of Oxford, 5 June 2018, Oxford (UK).

Abstract

The present work is a collection of articles [133, 134, 135] that, in broad terms, are dedicated to the development of statistical models for the analysis of social networks. Two main approaches are adopted throughout this manuscript that, despite being substantially different in their nature, are also complementary and accompanied by an active and relevant scientific background.

From one side, following the statistical physics literature regarding the study of networks, we develop models based on the topology of the observed graphs: these methods are built starting from the activity of each node, i.e. its degree, and this information is preserved on average in the benchmark structure. Therefore any successive analysis that compares observed and expected quantities to detect relevant behaviours is, indeed, identifying quantities that cannot be simply reproduced by a model built on the information regarding solely network topology.

On the other side, we also analyse the stream of literature that focuses on generative models. In this framework, the network topology plays a role in the definition of the temporal evolution of the node-related link probability, synthesized in the preferential attachment rule that is a function of nodes' degrees. We enrich this formulation in order to take into account also some individual features of the nodes, not related to the network structure. This addition gives a fundamental contribution in reproducing the evolution of the considered network.

Which of the two approaches provides a more accurate benchmark is often argument of debate. In this work we have developed theoretical tools and tested them with real-world applications, for both the presented cases. The results of our analyses show that the information provided by these two complementary methodologies can reveal equally valuable in reproducing different aspects of the systems object of study.

Chapter 1

Introduction to Complex Networks

In nature, many observable phenomena can be described starting from the interactions among a collection of objects. In many cases the objects involved may be numerous and these existing relations are often non-trivial. Just consider the examples of the Internet or the World Wide Web [1, 2, 3, 4, 5]: in this framework, the objects are represented by a huge amount of documents while their interactions are hyperlinks connections between one document and another. Another example is the brain [6, 7], that represents the set of interactions among a large number of neurons through synapses. Because of their structural “complexity”, these natural systems are often described with the term *complex systems* and the research of appropriate tools to analyse them has been increasing and developing since the last century. In this regard, the use of *graphs* as skeletons [8] for these systems has become popular and successful and the behaviour of many complex systems has been modeled and studied by means of these mathematical instruments.

One of the branches of the complex systems community has focused its attention in employing network tools with the purpose of modeling the social interactions and interpersonal relationships within populations of individuals. These kind of relations may have different ori-

gins. For instance, one may be interested in analysing the friendship relations within a social network context [9], or the collaborations among the group of scholars in academia [10, 11, 12, 13], or the collaborations among actors that work together when filming a movie [14, 15]. The areas of the academic research focused on the analysis of several aspects of the human behaviour and sphere are said to be devoted to the study of *social networks*. It has been argued that social networks are different with respect to the other types of networks [16], because of their overall architecture and the properties of the relations connecting individuals. The rest of this work will be mostly dedicated to the analysis of this kind of systems and each chapter will provide an ad-hoc introduction and description of the type of social network object of study.

It is common in literature to use the term “network” to denote the real-world phenomenon object of study, while the term “graph” is often used to indicate the mathematical object used to model the observed real-world network. The same rule will be followed (in general) also in this work, even though sometimes the terms will be mentioned interchangeably.

1.1 The basics

Consider the simple problem of representing the network of friendship relations among people belonging to the same social group. This situation can be easily represented as a graph $G = (\mathcal{N}, \mathcal{E})$, being \mathcal{N} the set of individuals in the group and $\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$ the set of pairwise friendship interactions. Therefore, the generic object $(i, j) \in \mathcal{E}$ indicates a friendship relation between individuals i and j .

However, in practice there always exists a gap between the *real* set of friendship relations among these people and the way in which these connections are *represented* with a network fashion. Indeed there are several possible approaches to detect the friendship relations among a set of agents. One possibility could be to check who is friend with whom on some online social network (e.g. Facebook). If two users are friends on Facebook, then we can (presumably) infer that they are also friends

in the real life. Another possibility could be asking each of them to list the names of their friends; in this case we may observe asymmetric relations, as shown in [9]. Indeed, the fact that i perceives j as one of her/his friends does not necessarily imply that j has also mentioned her/him. For instance in [9] some students from US high and junior-high schools are asked to list the names of their friends and the authors observe that most of the links in which lower-classes students mention higher-classes ones as their friends are non-reciprocated. Finally, one might employ some additional information, for instance verify if any of these people go to the same school, or practice to the same gym, or attend the same music class. In this latter case, we may consider friends all people attending the same music class, or going to the same school or practicing to the same gym. Then, the network of friendships is built simply adding a friendship relation $(i, j) \in \mathcal{E}$ whenever we observe i and j to be connected in one of the previously listed ways.

Notice that any of these procedures would lead to a completely different set of pairwise interactions. The element $(i, j) \in \mathcal{E}$ in the first approach indicates that i and j are friends on Facebook. A simple example of graph resulting from this construction method is shown in top-left panel of Figure 1, where the grey circles denote the considered users. All edges are reciprocated, since if i is friend with j on Facebook, it is not possible that j decides not to be friend with i . This kind of graphs are *undirected*.

On the contrary, non-reciprocated edges are possible with the second approach, since in this case $(i, j) \in \mathcal{E}$ means that i has listed j as one of her/his friends, thus it does not necessarily imply that also j has listed i as one of her/his friends. Therefore $(i, j) \in \mathcal{E}$ does not imply $(j, i) \in \mathcal{E}$. A real-world example of this situation is provided in [9], where the authors show that unreciprocated friendships are often motivated by reasons related to the social role covered by the individuals in the considered context: in most of the cases, people from lower classes claim friendship relations with people belonging to higher classes at school. This kind of graphs are *directed* and are graphically represented in top-right panel of Figure 1 where the direction of the arrow indicates who has listed whom

as her/his friend.

Finally, in the last approach the network of friendships is built starting from an auxiliary information: the edge $(i, j) \in \mathcal{E}$ actually indicates that i and j do some kind of activity together, either they go to the same school, or practice to the same gym, or attend the same music class, just to mention a couple of examples. Thus we preliminary represent a network of users and (for instance) school classes, as shown in bottom panel of Figure 1. Gray circles indicate agents while black squares represent school classes: two users being connected to the same black square means that they attend one class together at school. Therefore we simply *infer* that they are likely to be friends with each other and include (i, j) in the set of friendship relations \mathcal{E} . Graphs as the one represented in bottom panel of Figure 1 are called *bipartite*. As in the simple friendship example, bipartite networks involve two different types of nodes and no edges between nodes of the same type are observed.

For the sake of simplicity, we have considered friendship as a *binary* relation, meaning that i can only “be” or “not be” friend with j and it is not possible to observe that i is “less” or “more” friend with j than with somebody else. When this possibility is considered, an edge indicating friendship between i and j is accompanied by a weight quantifying the *intensity* of their relation. Such kind of networks are called *weighted*. The rest of this work will mainly deal with directed or undirected bipartite networks. Thus, in what follows I will provide a more formal definition of these kinds of graphs and of the most important quantities employed to describe them.

1.1.1 Adjacency matrix

The first element that needs to be introduced is the *adjacency matrix* of a graph. Any undirected network $\mathbf{G} = (\mathcal{N}, \mathcal{E})$ admits an equivalent representation in terms of its adjacency matrix $\mathbf{A} = \{a_{i,j} : i, j \in \mathcal{N}\}$ with binary entries

$$a_{i,j} = \begin{cases} 1 & \text{if } (i, j) \in \mathcal{E} \\ 0 & \text{otherwise.} \end{cases} \quad (1.1)$$

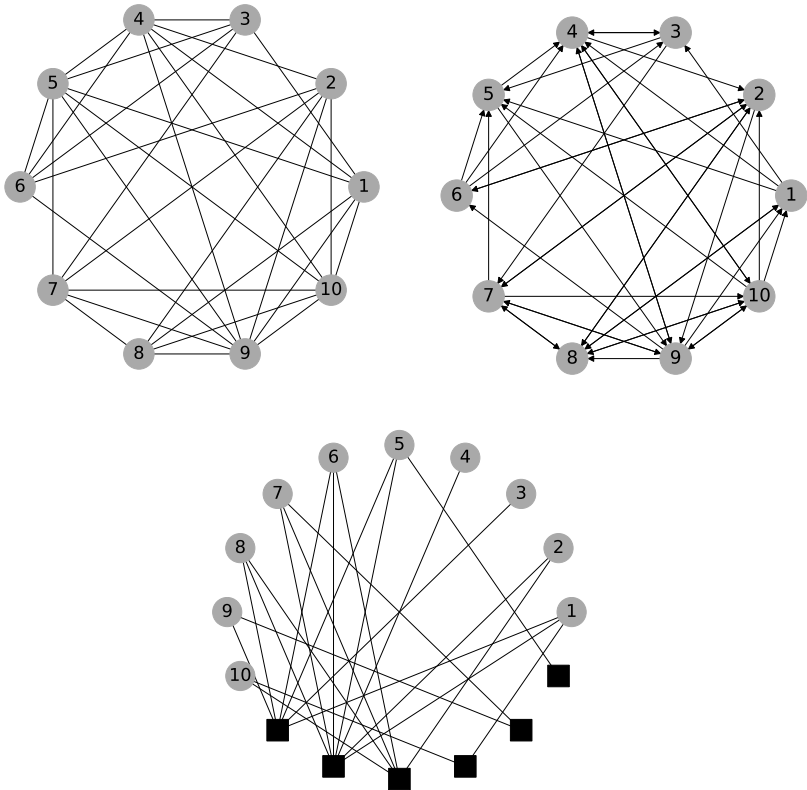


Figure 1: Graphical representation of simple examples of undirected, directed and bipartite graphs. The top-left panel shows reciprocated relations between individuals; the top-right panel shows directed edges, therefore possibly non-reciprocated relations; the bottom panel shows relations involving two different sets of units in the network.

Matrix \mathbf{A} has shape $N \times N$ where $N = \text{card}(\mathcal{N})$ is the total number of nodes in the network, while its number of edges $E = \text{card}(\mathcal{E})$ is given by

$$E = \sum_{i \leq j} a_{i,j}.$$

Indicating with $\mathcal{N}^* = \{1, \dots, 10\}$ the set of friends and with \mathcal{E}^* the set of friendship relations (in what follows we will distinguish the notation according to the mechanism used to collect the friendship interactions) the adjacency matrix representing the undirected friendship network $\mathbf{G}^* = (\mathcal{N}^*, \mathcal{E}^*)$ in top-left panel of Figure 1 is the following

$$\mathbf{A}^* = \begin{pmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \end{pmatrix}$$

The asterisk $*$ notation is used whenever dealing with an *observed* graph, as in this case, and will be adopted hereafter. In other words, any entry $a_{i,j}^*$ is equal to 1 whenever users i and j are friends with each other in graph \mathbf{G}^* and 0 otherwise. Notice that \mathbf{A}^* is symmetric, since the edges in the network of Facebook friendships are reciprocated. Therefore, if both $(i, j) \in \mathcal{E}^*$ and $(j, i) \in \mathcal{E}^*$ then $a_{i,j}^* = a_{j,i}^* = 1$. Moreover, all elements along the diagonal are set equal to zero since no user can be “friend with her/himself”. Networks of this kind are said to have *no self-loops*. Even when not directly specified, all applications of this work will consider networks without self-loops.

Consider instead the directed network of friendship relations $\mathbf{G}_{\text{Di}}^* = (\mathcal{N}^*, \mathcal{E}_{\text{Di}}^*)$ introduced in top-right panel of Figure 1. The friendship adja-

cency matrix induced by this graph is the following

$$\mathbf{A}^* = \begin{pmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}.$$

In this case the entry $a_{i,j}^* = 1$ indicates that i has listed j as one of her/his friends, thus the resulting matrix may not be symmetric, since $(i, j) \in \mathcal{E}_{\text{Di}}^*$ does not necessarily imply that $(j, i) \in \mathcal{E}_{\text{Di}}^*$. As in the undirected case, the adjacency matrix of a directed network has shape $N \times N$, while the total number of edges in the graph can be recovered as

$$E = \sum_{i \neq j} a_{i,j}.$$

1.1.2 Bipartite networks

Consider the last example introduced in the previous section. In this case the friendship network has been inferred starting from a different structure, the network of users and school classes, where a link connecting a student and a class indicates that the girl/boy attends that course at school. This kind of graphs are called *bipartite*. In a bipartite network, the set of nodes \mathcal{N} can be partitioned in two subsets of vertices $L = \{1, \dots, N_L\}$ and $\Gamma = \{1, \dots, N_\Gamma\}$ called *layers* and only edges connecting nodes belonging to different layers are possible. In what follows, Latin and Greek letters will be respectively used to index nodes on the first and second layer of the network. Any bipartite, undirected graph $\mathbf{G}_{\text{Bi}} = (L, \Gamma, \mathcal{E}_{\text{Bi}})$ can be represented by means of its *biadjacency matrix* $\mathbf{M} = \{m_{i,\alpha} : i \in L, \alpha \in \Gamma\}$ with binary entries

$$m_{i,\alpha} = \begin{cases} 1 & \text{if } (i, \alpha) \in \mathcal{E}_{\text{Bi}} \\ 0 & \text{otherwise.} \end{cases} \quad (1.2)$$

Matrix \mathbf{M} has shape $N_L \times N_\Gamma$ and, as in its monopartite counterpart, the total number of edges can be recovered as

$$E = \sum_{i,\alpha} m_{i,\alpha}.$$

Indicating again with $L^* = \{1, \dots, 6\}$ the set of classes at school and with $\Gamma^* = \{1, \dots, 10\}$ the set of friends, the bipartite network of school-mates and classes $\mathbf{G}_{\text{Bi}}^* = (L^*, \Gamma^*, \mathcal{E}_{\text{Bi}}^*)$ reproduced in bottom panel of Figure 1 has the following biadjacency matrix

$$\mathbf{M}^* = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

where each entry $m_{i,\alpha}^* = 1$ if user i attends class α and $m_{i,\alpha}^* = 0$ otherwise. At this point, in order to reconstruct the network of friendship relations between individuals, we take into account the following quantities

$$\sum_{\alpha} m_{i,\alpha}^* m_{j,\alpha}^* \quad \forall i, j \in \mathcal{N}$$

that simply indicate the number of classes that i and j attend together. Whenever this quantity is greater than zero, nodes i and j attend at least one class together and the edge (i, j) is added to the set of edges \mathcal{E}^* of the network of friendship, since we consider likely that i and j can be friends with each other.

The notation introduced in this paragraph will be used also in the following sections. I will use the terms $\mathbf{G} = (\mathcal{N}, \mathcal{E})$ for undirected and monopartite networks, while the simple generalizations $\mathbf{G}_{\text{Di}} = (\mathcal{N}, \mathcal{E}_{\text{Di}})$ and $\mathbf{G}_{\text{Bi}} = (L, \Gamma, \mathcal{E}_{\text{Bi}})$ will be used respectively for directed and bipartite networks. In general, the names used for the adjacency matrices will be chosen in each application, in order to maximise the clarity of the expressions.

1.1.3 Nodes' degrees

Given any undirected network \mathbf{G} and its adjacency matrix \mathbf{A} , the number of edges incident to node i can be recovered by means of the following quantity

$$k_i = \sum_{j \in \mathcal{N}} a_{i,j} \quad \forall i \in \mathcal{N}.$$

The term k_i is the *degree* of node i and is obtained as the sum of the elements in the i -th row (or column) of the adjacency matrix. For the undirected case in the friendship example, this quantity can be easily interpreted as the number of friends that user i has on Facebook. Instead, when considering directed networks, the two following similar quantities can be considered

$$\begin{aligned} k_i^{out} &= \sum_{j \in \mathcal{N}} a_{i,j}, \quad \text{and} \\ k_i^{in} &= \sum_{j \in \mathcal{N}} a_{j,i} \quad \forall i \in \mathcal{N} \end{aligned}$$

respectively called *out-degree* and *in-degree* of node i and indicating the number of out-going and in-going links incident to node i . In terms of the friendship network introduced in the previous section, such quantities respectively represent the number of users listed by i as her/his friends and the number of users who have listed i as one of their friend. Finally, when dealing with bipartite networks, the definition of node degrees can be easily extended as follows

$$\begin{aligned} k_i &= \sum_{\alpha \in \Gamma} m_{i,\alpha} \quad \forall i \in L \\ k_\alpha &= \sum_{i \in L} m_{i,\alpha} \quad \forall \alpha \in \Gamma \end{aligned}$$

specifying a degree sequence for each layer in the network. Their interpretation in the friendship example is analogous to the previous ones: the degree of node i indicates the number of classes attended by i , while the degree of α denotes the number of students enrolled in class α .

Notice that the following sequence of equalities involving node degrees holds for simple networks

$$E = \frac{1}{2} \sum_{i \neq j} a_{i,j} = \frac{1}{2} \sum_{i \in \mathcal{N}} k_i$$

while it straightforwardly translates to the cases of directed and bipartite networks as follows

$$E = \sum_{i \neq j} a_{i,j} = \sum_{i \in \mathcal{N}} k_i^{out} = \sum_{i \in \mathcal{N}} k_i^{in}$$

and

$$E = \sum_{i,\alpha} m_{i,\alpha} = \sum_{i \in L} k_i = \sum_{\alpha \in \Gamma} k_\alpha.$$

1.1.4 Network quantities

Whenever studying a real-world system, the goal is to understand its structure at a global level and to study the relations intervening among its parts. In this regard, there are some fundamental quantities that need to be analysed, providing information both at a global and local level of the graph. For the sake of simplicity, only the definition related to the monpartite, undirected case will be provided, since in most of the cases the extension to the bipartite or directed setting is simply straightforward.

Paths

Two different edges that are incident to the same vertex are called *consecutive*. Therefore, a *path* is a sequence of consecutive edges connecting a pair of nodes in the network. The *length* of a path is the number of consecutive edges in the path.

Distance

Given two nodes of a graph $i, j \in \mathcal{N}$ we define the *distance* between them $\ell(i, j)$ as the length of the shortest path connecting i and j . The

assumption $\ell(i, j) = +\infty$ is used whenever node i cannot be reached from node j . It is worth to notice that the concepts of length and distance are intended as topological quantities: they simply refer to the required number of edges that need to be walked on in order to move along the whole path, they have nothing to do with the real geographical distance between i and j .

Diameter

Given $\mathcal{D} = \{\ell(i, j) : i, j \in \mathcal{N}\}$ the set of distances between any pair of nodes in the network, the *diameter* of the network can be computed as $D = \max(\mathcal{D})$ and represents the distance between the two farthest nodes in the network.

Average shortest-path-length

The *average shortest-path-length* of a connected network is computed as

$$\langle \ell \rangle = \frac{2}{N(N-1)} \sum_{i,j \in \mathcal{N}} \ell(i, j)$$

and represents the average value of the distances between any pair of nodes in the network.

Connectivity

A graph is *connected* whenever there exists a path connecting any pair of nodes in the network. As a consequence, a *connected component* of a graph is a set of vertices such that any pair of nodes can be connected by a path. I will use the notation ν to indicate the number of connected components of a graph.

Connectance

The *connectance* of a graph d is defined as

$$d = \frac{2E}{N(N-1)}$$

and simply represents the ratio between the number of actually observed edges E and the total number of possible ones $\binom{N}{2}$.

Clustering

The *clustering coefficient* for node $i \in \mathcal{N}$ is defined as

$$C_i = \frac{2\tau_i}{k_i(k_i - 1)} = \frac{\sum_{j \neq i} \sum_{k \neq i, j} a_{i,j} a_{j,k} a_{i,k}}{\sum_{j \neq i} \sum_{k \neq i, j} a_{i,j} a_{i,k}} \quad (1.3)$$

where the term τ_i represents the number of closed triangles in which node i is involved. This means that the previous quantity indicates the ratio of i 's neighbours that are themselves connected to each other. Starting from this definition, the *average clustering coefficient* C of the whole graph is obtained as

$$C = \frac{1}{N} \sum_{i \in \mathcal{N}} C_i$$

averaging the values C_i overall the set of nodes in the network. A network with high clustering C has a high percentage of closed triplets, therefore two neighbours of the same nodes have a high chance to be neighbours themselves.

Assortativity

The *assortativity* coefficient s measures the correlation between degrees of pairs of connected nodes. s assumes values within the interval $[-1, 1]$: $s = 1$ indicates assortative patterns, links are observed between nodes of similar degrees; $s = -1$ indicates disassortative patterns, so connections are observed mostly between nodes with different degrees; while $s = 0$ indicates absence of correlation.

An alternative way to analyse the correlations between node degrees is studying the trend observed in the *average nearest-neighbour-degree*. This quantity is defined as

$$k_i^{nn} = \frac{\sum_{j \neq i} a_{i,j} k_j}{k_i} = \frac{\sum_{j \neq i} \sum_{k \neq j} a_{i,j} a_{j,k}}{\sum_{j \neq i} a_{i,j}} \quad \forall i \in \mathcal{N} \quad (1.4)$$

and indicates the average value of the degrees shown by i 's neighbours. Averaging this quantity with respect to all nodes in the network having the same degree and plotting its trend, it is possible to understand the assortativity patterns: if the trend is increasing the network is assortative, if the trend is decreasing the network is disassortative.

Among the previously listed quantities, the most studied in the scientific literature are probably the clustering and assortativity coefficients. The reason becomes clear if we simply consider their interpretation in the case of the friendship network: a high clustering coefficient related to person i indicates that a high fraction of i 's friends are also friend with each other, while an increasing (or decreasing) assortativity pattern denotes that there is a positive (negative) correlation between the degrees of the people, i.e. (non) popular guys tend to be friends with other popular guys. In both cases, these quantities provide important information regarding the topological structure of the friendship network.

In this first section we introduced the most important, although essential, quantities related to the study of real-world social network. The remainder of this Chapter will be mainly devoted to the introduction of the most well-known and studied network models.

1.2 Network models

The interest in network models finds its roots in the field of graph theory during the mid years of the last century, when the mathematicians Paul Erdős and Alfréd Rényi developed the *Random Graph* model. According to it, starting from an initial set of disconnected nodes, each pair of vertices is connected at random with fixed probability [17, 18, 20]. Their proposal has been considered of crucial importance in the field and the properties of the networks generated with such a mechanism have been largely studied.

Nevertheless, the choice of the Erdős-Rényi model needed to be reconsidered since many observable properties of real-world networks sub-

stantially deviate from its expectation, the most remarkable example being the distribution of node degree in a (sufficiently large) network, that is approximately Poisson distributed in a Random Graph. In contrast, many empirical findings show that real-world networks such as the Internet [1, 4, 5] or the World Wide Web [2, 3] or metabolic networks [20, 21] have a degree distribution that is Exponentially or Power-Law distributed and therefore significantly deviate from the random behaviour. Those networks are called *scale-free* because of the property of scale invariance of their degree distribution [19, 22]. For this reasons, much effort has been devoted to the development of network models that could reproduce the real networks behaviour. This section proposes a review of the network models literature, presenting the following ones: Random Graph, Barabási-Albert, Fitness and Exponential Random Graph.

1.2.1 Erdős-Rényi Random Graph

In the seminal Erdős-Rényi Random Graph, N nodes are connected by E edges drawn uniformly at random from the set of $\binom{N}{2}$ possible edges. Therefore, a simple null model for a real-world network can be defined simply considering the two following components:

- an *ensemble of graphs* \mathcal{G} , defined as the set of all graphs with N nodes and E edges, and
- a *probability distribution over the ensemble*, essentially representing the probability to observe each graph in the set \mathcal{G} .

The number of networks in such an ensemble can be obtained computing all the possible ways of drawing E edges among a set of $\binom{N}{2}$ possible ones. With this definition, all the network configurations are equiprobable and this probability is equal to

$$P(\mathbf{G}) = \left(\frac{\binom{N-1}{2}}{E} \right)^{-1} \quad \text{for } \mathbf{G} \in \mathcal{G}.$$

However, the most studied version of the Erdős-Rényi Random Graph does not fix the number of edges a priori and is also known as *binomial model* [17, 18, 20, 23]. It prescribes to start from an initial set of N nodes and each of the $\binom{N}{2}$ possible edges between them is independently drawn with fixed probability $p \in (0, 1)$. Notice that the two extreme cases $p = 0$ and $p = 1$ have been excluded since they would generate the trivial situations of fully disconnected and fully connected graphs, respectively. More formally, using the notation introduced in Section 1.1 and Subsection 1.1.4 for a generic graph $\mathbf{G} \in \mathcal{G}$, each entry $a_{i,j}$ of \mathbf{G} 's adjacency matrix is treated as an independent, Bernoulli-distributed random variable with parameter

$$\begin{aligned} P(a_{i,j} = 1) &= p \\ P(a_{i,j} = 0) &= 1 - p \end{aligned} \tag{1.5}$$

for $i, j = 1, \dots, N$. Therefore, the total number of edges in the network $E = \sum_{i < j} a_{i,j}$ is a Binomial random variable with expected value in the ensemble of graphs \mathcal{G} given by the sum of the expected values of each binary variable $a_{i,j}$, i.e.

$$\langle E \rangle = \sum_{i < j} \langle a_{i,j} \rangle = \frac{N(N-1)}{2} p. \tag{1.6}$$

Thus, the probability to “pick” each of the graphs in the ensemble can be expressed as a function of the total number of present links. For instance, each graph $\mathbf{G} \in \mathcal{G}$ with \bar{E} edges is observed with probability

$$P(\mathbf{G}|\bar{E}) = p^{\bar{E}}(1-p)^{\frac{N(N-1)}{2} - \bar{E}}.$$

Degree distribution in Erdős-Rényi Random Graph

In the Random Graph model, each link of the network is treated as an independent and Bernoulli-distributed random variable with fixed parameter $p \in (0, 1)$. As a consequence, using again the notation introduced in Section 1.1 and Subsection 1.1.4, the degree of node $i \in \mathcal{N}$, $k_i = \sum_{j \in \mathcal{N}} a_{i,j}$ is a Binomial random variable and assumes value k whenever k of the possible edges with the remaining $N - 1$ vertices are

present in the system, while all the other $N - 1 - k$ are not realized. Since there are $\binom{N-1}{k}$ possible ways of selecting k neighbours from a set of $N - 1$ nodes, the degree distribution for node i is the following

$$P(k_i = k) = \binom{N-1}{k} p^k (1-p)^{N-1-k} \quad (1.7)$$

with $0 \leq k \leq N - 1$. At this point, the probability that a node picked at random has degree k can be derived considering the random variable X_k representing the number of nodes in the network having degree k (the notation is borrowed from [20]). It has been shown in [20] that X_k has approximately a Poisson distribution

$$P(X_k = \kappa) = e^{-\lambda_k} \frac{\lambda_k^\kappa}{\kappa!} \quad (1.8)$$

with $0 \leq \kappa \leq N$ and expected value, derived from equation (1.7), equal to

$$\langle X_k \rangle = NP(k_i = k) = \lambda_k.$$

However, the authors in [20, 24] show that the distribution in equation (1.8) does not show significant deviations from its average value. Thus, with a good accuracy, the degree distribution of a Random Graph can be approximated with the Binomial distribution

$$P(k_i = k) \approx \frac{\langle X_k \rangle}{N} = \binom{N-1}{k} p^k (1-p)^{N-1-k}$$

that, for sufficiently large N , converges to the Poisson distribution

$$P(k_i = k) \approx e^{-pN} \frac{(pN)^k}{k!} = e^{-\langle k \rangle} \frac{\langle k \rangle^k}{k!}$$

that represents the probability that a node picked at random from the set of vertices has degree k . The previous equation highlights that the majority of the nodes in an Erdős-Rényi Random Graph has a degree approximately equal to the expected value $\langle k \rangle = Np$ while more extreme values of the degrees are less likely to appear. The result is graphically represented in Figure 2: the three presented distributions have different average degrees, $\langle k \rangle = 2$ for the blue squares, $\langle k \rangle = 5$ for the orange circles and $\langle k \rangle = 10$ for the green stars. In all these cases the highest probability is assigned to degree values close to the average $\langle k \rangle$.

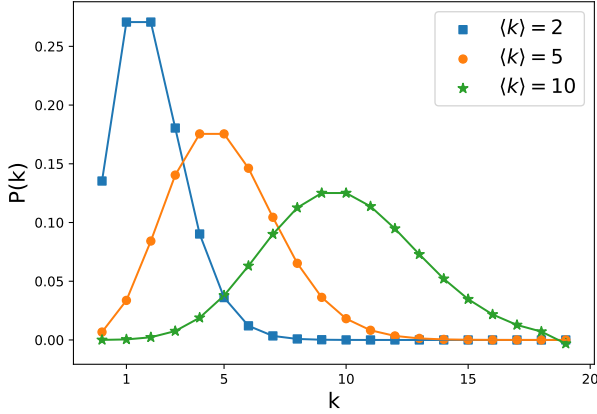


Figure 2: Three examples of Poisson degree distributions with different average degree $\langle k \rangle$. The blue squares indicate $\langle k \rangle = 2$, the orange circles are with $\langle k \rangle = 5$ while the green stars denote $\langle k \rangle = 10$. Notice that the majority of nodes shows a degree value close to the average one.

1.2.2 Barabási-Albert model

As stated in the previous Section 1.2, many real-world networks are characterized by a degree sequence that is Exponentially or Power-Law distributed, therefore significantly different from the shape displayed by random graphs. For this reason, much effort has been developed in trying to understand what is the driving mechanism for this phenomenon.

The problem was first addressed by [19, 20, 22]. Their model is based on two fundamental ingredients that drive the formation of a network with Power-Law degree distribution: first, the network is not assumed to have a fixed dimension but it grows in time with the addition of new nodes at every time step; second, the new edges between nodes are not placed at random but the probability that a new edge is connected to an already existing node depends on the degree of the existing node itself. Given these two driving rules, the algorithm develops according to the following steps:

1. The model starts at time $t_1 = 1$ with an initial set of N_1 nodes, randomly connected by E_1 edges. In order to obtain no isolated nodes, it is necessary that each vertex has at least one link, to have a positive probability of connection for all nodes.
2. At every timestep $t \geq 2$ a new node is added with a fixed number of new edges e that connects it to the other vertices already present in the system. Therefore, at timestep t , the network will be composed of N_t nodes and E_t edges where

$$\begin{aligned} N_t &= t + N_1 \\ E_t &= E_1 + et \end{aligned} \tag{1.9}$$

since only one new node with a fixed number of edges e is added at every time-step. Extending the notation introduced in Section 1.1 to the time-dependent formalism, in what follows the term \mathcal{N}_t is used to indicate the set of nodes present in the system at time t while $k_{i,t}$ is used for the degree of node i at time t .

3. The probability of connecting new-entry nodes to the already existing ones is regulated by a mechanism called *preferential attachment*, that works as follows: the probability of a new-entry node to be connected to an already existing node $i \in \mathcal{N}_{t-1}$ is directly proportional to i 's degree until time $t - 1$. In symbols

$$p_{i,t} = \frac{k_{i,t-1}}{\sum_{j \in \mathcal{N}_{t-1}} k_{j,t-1}}. \tag{1.10}$$

Equation (1.10) states that nodes with larger degrees have a higher probability to be connected to the newcomers, thus they are also likely to become *hubs* in the network. On the contrary nodes with lower degrees are likely to remain less connected. This mechanism is also known with the term *rich-gets-richer*. Both numerical simulations and analytical results show that the network evolves in such a way that the final degree sequence has approximately a Power-Law degree distribution with parameter close to $\gamma = 3$

$$P(k_i = k) \propto k^{-\gamma}$$

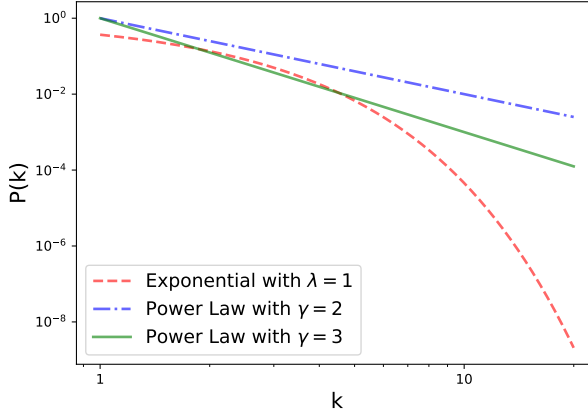


Figure 3: Comparison between many real-world networks degree distribution, that show a Power Law behaviour (the blue dotdashed line is obtained with $\gamma = 2$, while the green and continuous one is characterised by $\gamma = 3$) and the Exponential degree distribution (in orange, with $\lambda = 1$).

as shown in [19]. A graphical representation of this behaviour is provided in Figure 3. The blue dotdashed line represents a Power-Law distribution with parameter $\gamma = 2$, while the green continuous line is obtained with $\gamma = 3$. Both cases are compared with the orange dashed line, representing an Exponential distribution with parameter $\lambda = 1$. Clearly, the main difference between these distributions is related to their behaviour for large degrees, since the Exponential decay is much faster than the one observed for the Power-Law case.

Example. A toy example of the Scale-Free model is proposed in Figure 4. In this case the initial configuration is made by the set $\mathcal{N}_1 = \{1, 2, 3\}$ of $N_1 = 3$ nodes and $E_1 = 3$ edges, thus all nodes have the same probability of attracting new neighbours equal to $p_{i,2} = 1/3$ for all $i = 1, 2, 3$. When node 4 joins the system at time $t = 2$ with $e = 1$ new edges, it chooses uniformly at random node 1 as neighbour. Therefore node 1 acquires the highest probability of gaining new neighbours equal to $p_{1,3} = 3/8$

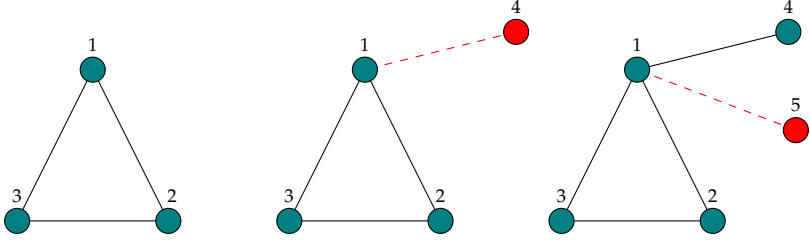


Figure 4: The Figure shows three consecutive timesteps of the Scale-Free model. In this case the initial set is made of $N_1 = 3$ nodes and $E_1 = 3$ edges. At each timestep a new node enters the system showing $e = 1$ new edges. New nodes are depicted in red while old ones in dark green. The probability to form a link with the existing nodes is increasing with the degree of the nodes already present in the system.

contrarily to the other nodes with $p_{2,3} = p_{3,3} = 2/8$ and $p_{4,3} = 1/8$. The same procedure is repeated at time $t = 3$. So, with the arrival of node 5, the probabilities to form links with the newcomers become $p_{1,4} = 2/5$, $p_{2,4} = p_{3,4} = 1/5$ and $p_{3,4} = p_{4,4} = 1/10$.

Connection with urn processes

An interesting property of the preferential attachment rule is that it can be explained with an urn process. Indicate with t_i the arrival time of node $i \in \mathcal{N}_{t-1}$ in equation (1.10), with $1 \leq t_i \leq t - 1$ and suppose to have an urn with balls of two colours, say red and black. The initial number of balls in the urn is $k_{i,t_i} + \sum_{j \in \mathcal{N}_{t_i-1}} k_{j,t_i-1}$, while the number of black balls in the urn depends on the degree of node i at the time of its arrival: if $t_i = 1$ then i is one of the N_1 nodes present in the system at time $t_1 = 1$, thus the number of black balls in the urn equals the degree of node i at time $t_i = 1$ (that, in turns, depends on the configuration of the E_1 edges present in the system at time $t = 1$). If instead $t_i > 1$ then the number of black balls in the urn is equal to e , i.e. the number of new edges brought in the system by node i at the time of its arrival. Then, perform an extraction from the urn for each $t > t_i$. If the drawn ball is black then node t establishes a link with node i and an additional black

ball enters the urn. On the contrary, if the extracted ball is red, node t does not establish a link with i and another red ball is added to the urn. Thus, equation (1.10) represents the probability to extract a black ball at timestep t .

Further developments of the model

The preferential attachment mechanism was already present in the scientific literature but in a slightly different form [25, 26]. However, the Barabási-Albert model boosted the attractiveness of complex networks and many other scholars delved into the investigation of the properties of generative models (nice reviews on the subject can be found in [27, 28]).

However, the Scale-Free model is known to suffer of the “first-move advantage”, i.e. the fact that older nodes have a greater degree by construction. Thus, in subsequent contributions, its recipe was enriched with other ingredients in order to face this issue. One of the most remarkable ones are the *fitness variables*: the fitness is a quantity defined at node level that measures the intrinsic ability of the vertex to collect links. These quantities have been introduced in [29, 30] with a multiplicative effect on nodes’ degrees, with the purpose of amplifying or dampening the preferential attachment effect. Therefore the presence of the fitnesses in the model permits to overcome the first-move advantage and allows younger nodes to easily grow as well.

In addition to the mentioned contributions, some other works have been proposed that try to relate the scale-free behaviour of the systems (i.e. the power-law degree distribution) with different generative mechanisms involving the fitnesses. In [31] the authors assign a fitness variable to every node, representing an intrinsic property of the vertex in the considered system. Then, arcs between pairs of vertices are drawn with a probability that depends on the fitnesses of the involved nodes only. Therefore, also the scale-free behaviour is explained as a result of the fitnesses’ contribution only. The following section describes the generative model presented in [31].

1.2.3 Fitness model

As stated in the previous paragraph, the generative network model presented in [31] represents the first contribution in which the scale-free behaviour of a system is explained in terms of the importance of each node and not as a function of its popularity (i.e. preferential attachment rule). The dynamical version of the model works as follows:

1. Start with an initial set of N_1 nodes. Each node i of the system is assigned a fitness variable x_i , a real number representing the importance of such a node in the system. In this framework, the fitnesses are treated as i.i.d. random numbers, distributed according to a given probability distribution that we denote with ϕ .
2. The set of edges at time $t = 1$ is determined as follows: for each pair of nodes (i, j) the presence of a link connecting them is established with probability

$$p_{i,j} = f(x_i, x_j). \quad (1.11)$$

Essentially, the presence of a link between i and j is determined as a function of the importance of both nodes, measured according to the values of their fitness variables.

3. At every timestep $t \geq 2$ a new set of nodes is added to the system, each of them accompanied by its fitness variable. Edges between old and new vertices are again established according to the rule in equation (1.11).

The authors in [31] show that the choice of power-law or exponentially distributed fitnesses, that correspond to the cases $\phi(x) \sim x^{-\gamma}$ with $\gamma \geq 2$ and $\phi(x) \sim e^{-x}$, leads to the generation of scale-free networks. In this case, the simulations and computations have been performed with a threshold rule as link probability, i.e. $p_{i,j} = \Theta[x_i + x_j - z]$. The term Θ is the Heaviside step function, equal to one whenever its argument is greater than zero and equal to zero otherwise. Essentially, an arc between nodes i and j exists only when the sum of their fitnesses is greater than a fixed threshold, that in this case is equal to z .

The idea of importance of a node was already presented in the literature by [30]. The results in [31] strengthen its relevance simply excluding the “rich-get-richer” rule from the method. Moreover, the model presented here is dynamic and aimed at explaining the generation of simple networks only. However, its static formulation or its extension to the case of directed networks are trivial, the first one being obtained simply excluding step 3 from the algorithm. Other possible extensions may include time dependence in the definition of the fitnesses: for instance, the authors in [32] introduce fitnesses which values decay with time in order to take nodes’ ageing into account.

Besides generative models, nodes’ fitnesses can be employed to describe the structure of real networks also in other contexts, by correlating their value to some attributes of the nodes not directly specified in the definition of the network [31]. In the following section, inspired by [31], we will describe the Configuration Model methods, that follow this line of research. Also in this case, the fitnesses are defined at node level and are responsible for nodes’ degrees, but they are derived from an entropy maximization procedure. In the *Exponential Random Graph* framework, starting from the measurements of some properties on a real-world network (e.g. number of edges, clustering coefficient and so forth), the model is specified defining a suitable ensemble of graphs and a probability distribution over it. The aim is to obtain the probability distribution over the ensemble that identifies the networks that best reproduce the properties of interest [33].

1.2.4 Randomizing a real-world network

This section exploits the Exponential Random Graph Models framework in order to construct a randomized counterpart (i.e. the equivalent of a statistical null-model) for a specific real-world network of interest. Again, the asterisk $*$ notation will be used for observed quantities, such as the observed graph G^* or the observed adjacency matrix A^* . Given some properties of the observed network that we want to preserve in the randomized counterpart, we construct a suitable ensemble of possible net-

works and a probability distribution upon it, that satisfies (on average) the set of imposed constraints being however maximally random. The passages in this section are a collection of the results presented in [33, 34, 35, 36, 37].

Let \mathcal{G} be the ensemble of graphs consisting of all networks of the same type as \mathbf{G}^* (binary/weighted and/or directed/undirected) and with the same number of nodes N . Each graph $\mathbf{G} \in \mathcal{G}$ in the ensemble is assigned an occurrence probability $P(\mathbf{G})$ such that

$$\sum_{\mathbf{G} \in \mathcal{G}} P(\mathbf{G}) = 1. \quad (1.12)$$

The shape of the probability distribution over the ensemble $P(\mathbf{G})$ needs to be determined once a set of properties of the real-world network \mathbf{G}^* (i.e. constraints) are enforced in the model. Let \vec{X} be a vector of graph observables for which we have an empirical measurement in the real-network \mathbf{G}^* . We would like to obtain a probability distribution over the ensemble such that, for each observable X_i

$$\langle X_i \rangle = \sum_{\mathbf{G} \in \mathcal{G}} X_i(\mathbf{G}) P(\mathbf{G}) = X_i(\mathbf{G}^*) \quad (1.13)$$

the expected value of the quantities \vec{X} overall the ensemble coincides with the values of \vec{X} observed in the real-world graph \mathbf{G}^* . Here the notation $\vec{X}(\cdot)$ is used to indicate that the quantities \vec{X} are computed on the network in brackets. The desired probability distribution is the one that maximizes the Entropy

$$S = - \sum_{\mathbf{G} \in \mathcal{G}} P(\mathbf{G}) \ln P(\mathbf{G})$$

under the set of constraints in equation (1.13) and the normalization condition in equation (1.12). In order to do so, we first construct the Lagrange function introducing the Lagrange multipliers α and $\vec{\theta}$

$$L = S + \alpha \left(1 - \sum_{\mathbf{G} \in \mathcal{G}} P(\mathbf{G}) \right) + \sum_i \theta_i \left(\langle X_i \rangle - \sum_{\mathbf{G} \in \mathcal{G}} X_i(\mathbf{G}) P(\mathbf{G}) \right).$$

The partial derivative of L with respect to $P(\mathbf{G})$ is the following

$$\frac{\partial L}{\partial P(\mathbf{G})} = \ln P(\mathbf{G}) + 1 + \alpha + \sum_i \theta_i X_i(\mathbf{G})$$

for all graphs $\mathbf{G} \in \mathcal{G}$. Imposing the partial derivative equal to zero we obtain the solution of the maximisation problem

$$P(\mathbf{G}|\alpha, \vec{\theta}) = \exp \left(- (1 + \alpha) - \sum_i \theta_i X_i(\mathbf{G}) \right)$$

that is most commonly written as

$$P(\mathbf{G}|\vec{\theta}) = \frac{\exp \left(-H(\mathbf{G}, \vec{\theta}) \right)}{Z(\vec{\theta})} \quad (1.14)$$

where $H(\mathbf{G}, \vec{\theta})$ is the *Hamiltonian* of the problem

$$H(\mathbf{G}, \vec{\theta}) = \sum_i \theta_i X_i(\mathbf{G}) \quad (1.15)$$

and $Z(\vec{\theta})$ is a normalizing factor called *partition function*

$$Z(\vec{\theta}) = \exp(\alpha + 1) = \sum_{\mathbf{G} \in \mathcal{G}} \exp \left(-H(\mathbf{G}, \vec{\theta}) \right). \quad (1.16)$$

Equations (1.14)-(1.16) specify the Exponential Random Graph model, that consists in finding the probability distribution (conditional on the values of the parameters $\vec{\theta}$) over the ensemble of graphs \mathcal{G} that satisfies on average the set of constraints \vec{X} while keeping all the rest maximally random. Notice that we are dealing with a *grand-canonical* ensemble of graphs, meaning that the imposed constraints are satisfied *on average* overall the set of graphs and not “pointwise” on each graph $\mathbf{G} \in \mathcal{G}$. Therefore, graphs in which the constraints are not satisfied still belong to the ensemble.

At this point, the purpose of the analysis is the definition of the randomized counterpart of the *specific* real-world network \mathbf{G}^* . Thus the probability distribution $P(\mathbf{G}|\vec{\theta})$ has to be calibrated with the vector of Lagrange multipliers $\vec{\theta}^*$ that assign the highest probability to \mathbf{G}^* . In other

words, the probability distribution needs to be centered on \mathbf{G}^* . In order to do so we maximise the following log-likelihood function

$$\begin{aligned}\mathcal{L}(\vec{\theta}|\mathbf{G}^*) &= \ln P(\mathbf{G}^*|\vec{\theta}) \\ &= -H(\mathbf{G}^*, \vec{\theta}) - \ln Z(\vec{\theta})\end{aligned}\quad (1.17)$$

that represents the probability to obtain the network of interest \mathbf{G}^* . Equation (1.17) is maximised by the vector of parameters $\vec{\theta}^*$ ensuring that the average values in the ensemble of the quantities \vec{X} is equal to their empirical values on the network \mathbf{G}^* . Thus, for the Exponential Random Graph models, the maximum likelihood estimate of the parameters $\vec{\theta}$ guarantees that the enforced constraints on the observables \vec{X} are met in the ensemble:

$$\langle X_i \rangle_{\vec{\theta}^*} = \sum_{\mathbf{G} \in \mathcal{G}} X_i(\mathbf{G}) P(\mathbf{G}|\vec{\theta}^*) = X_i(\mathbf{G}^*) \quad (1.18)$$

for all i . As in [35], the notation $\langle \cdot \rangle_{\vec{\theta}^*}$ is used to indicate that the expected value in the ensemble is computed under the conditional probability distribution $P(\mathbf{G}|\vec{\theta}^*)$, calibrated on the maximum likelihood estimates of the parameters $\vec{\theta}^*$.

The constraints introduced in the problem \vec{X} are topological quantities of the real-world network, therefore they need to be established according to the nature of the network itself. For instance, for binary and undirected networks the degree of each node is used as a constraint: the second of the two following sections will present the Exponential Random Graph problem for this set of constraints. The first one represents an even simpler case, in which only the expected number of edges is preserved in the system: it shows that the Random Graph model can be addressed in the Exponential Random Graph framework imposing the total number of observed edges as a constraint of the problem.

Constraining on the number of edges: Random Graphs

Consider the simple network \mathbf{G}^* with fixed number of nodes N . The graph can be equivalently represented with its $N \times N$ adjacency matrix \mathbf{A}^* , with entries $a_{i,j} = 1$ if node i is connected to j and $a_{i,j} = 0$ otherwise.

Suppose to be interested in constructing a null model for such real-world network and to be interested in preserving the total number of edges E on average in the ensemble. The problem presented in the previous section needs to be specified for this particular case. The Hamiltonian of the problem with this single constraint reads as follows

$$H(\mathbf{G}, \theta) = \theta E(\mathbf{G}) = \theta \sum_{i < j} a_{i,j} \quad (1.19)$$

where θ is the real-valued Lagrange multiplier, while the partition function is

$$\begin{aligned} Z(\theta) &= \sum_{\mathbf{G} \in \mathcal{G}} \exp \left(-\theta \sum_{i < j} a_{i,j} \right) \\ &= \prod_{i < j} \sum_{\mathbf{G} \in \mathcal{G}} \exp(-\theta a_{i,j}) \\ &= \prod_{i < j} \left(1 + \exp(-\theta) \right)^{\binom{N}{2}}. \end{aligned} \quad (1.20)$$

Notice that the last steps are valid since the sum runs over all the binary entries of the adjacency matrix. Plugging equations (1.19) and (1.20) into the general expression (1.14) and using the substitution $\exp(-\theta) = x$, we obtain the following probability distribution over the graphs ensemble

$$\begin{aligned} P(\mathbf{G}|x) &= \prod_{i < j} \frac{\exp(-\theta a_{i,j})}{\left(1 + \exp(-\theta) \right)^{\binom{N}{2}}} \\ &= \prod_{i < j} \frac{x^{a_{i,j}}}{(1+x)^{\binom{N}{2}}} \\ &= x^E (1+x)^{-\binom{N}{2}} \\ &= \left(\frac{x}{1+x} \right)^E \left(1 - \frac{x}{1+x} \right)^{\binom{N}{2}-E} \\ &= p^E (1-p)^{\binom{N}{2}-E}. \end{aligned} \quad (1.21)$$

Notice that in equation (1.21) each entry of the adjacency matrix is treated as an independent and Bernoulli distributed random variable, with different probabilities of success equal to

$$p = \frac{x}{1+x}$$

therefore the probability associated to the entire graph is simply given by the product of these marginals. In order to obtain a maximum likelihood estimate for the parameter x , we need to maximize the log-likelihood function given by

$$\begin{aligned}\mathcal{L}(x|\mathbf{G}^*) &= \ln P(\mathbf{G}^*|x) \\ &= E(\mathbf{G}^*) \ln(x) - \binom{N}{2} \ln(1+x)\end{aligned}\tag{1.22}$$

and the result is the estimate $x^* > 0$ that guarantees that the enforced constraint on the expected number of links is met, on average, in the ensemble. The same solution x^* can be obtained solving the following equation

$$\frac{\partial \mathcal{L}(x|\mathbf{G}^*)}{\partial x} = \frac{E(\mathbf{G}^*)}{x} - \binom{N}{2} \frac{1}{1+x} = 0$$

that corresponds to the First Order Condition of the maximum likelihood problem. Rearranging the terms it is possible to obtain an analytical expression for the expected number of edges in the graph ensemble

$$E(\mathbf{G}^*) = \binom{N}{2} \frac{x^*}{1+x^*} = \langle E \rangle_{\theta^*}\tag{1.23}$$

since the term

$$p^* = \frac{x^*}{1+x^*} = \langle a_{i,j} \rangle_{\theta^*} \text{ for } i \neq j$$

represents the probability to observe a link between any pair of nodes in the system. Notice that the previous equation (1.23) represents a specific case for the general framework proposed in equation (1.18), in which the total number of edges in the graph is imposed as the only constraint. Again the notation $\langle \cdot \rangle_{\theta^*}$ is borrowed from [35] and represents the expected value in the ensemble of the quantity in parentheses computed

under the conditional probability distribution $P(\mathbf{G}|x^*)$ where $x^* = \exp(-\theta^*)$ is the maximum likelihood estimate of the Lagrangian multiplier of the problem. Notice also that the probability p is invariant for any edge. This result shows that the binomial model in Section 1.2.1 is actually an Exponential Random Graph model, obtained when the observed number of edges E is fixed as constraint in the graphs ensemble.

Constraining on the degree sequence: Generalized Random Graphs

Consider again the graph \mathbf{G}^* of the previous example, with fixed number of nodes N , E edges and adjacency matrix \mathbf{A}^* . In this case the interest is still in finding a suitable null model for the graphs \mathbf{G}^* , but preserving the degree observed for each node, i.e. the degree sequence of the network. The Hamiltonian of the problem becomes

$$H(\mathbf{G}, \vec{\theta}) = \sum_{i=1}^N \theta_i k_i(\mathbf{G}) = \sum_{i < j} (\theta_i + \theta_j) a_{i,j} \quad (1.24)$$

while the partition function is

$$\begin{aligned} Z(\vec{\theta}) &= \sum_{\mathbf{G} \in \mathcal{G}} \exp \left(- \sum_{i < j} (\theta_i + \theta_j) a_{i,j} \right) \\ &= \prod_{i < j} \sum_{\mathbf{G} \in \mathcal{G}} \exp \left(- (\theta_i + \theta_j) a_{i,j} \right) \\ &= \prod_{i < j} \left(1 + \exp(-\theta_i - \theta_j) \right). \end{aligned} \quad (1.25)$$

Again the last step holds since the sum runs over all the binary entries of the adjacency matrix. Plugging equations (1.24) and (1.25) into (1.14) and using the substitution $\exp(-\theta_i) = x_i$ for all $i \in \mathcal{N}$ we get the following

probability distribution over the graphs ensemble

$$\begin{aligned}
P(\mathbf{G}|\vec{x}) &= \prod_{i < j} \frac{\exp\left(-(\theta_i + \theta_j)a_{i,j}\right)}{1 + \exp(-\theta_i - \theta_j)} \\
&= \prod_{i < j} \frac{(x_i x_j)^{a_{i,j}}}{1 + x_i x_j} \\
&= \prod_{i < j} \left(\frac{x_i x_j}{1 + x_i x_j}\right)^{a_{i,j}} \left(1 - \frac{x_i x_j}{1 + x_i x_j}\right)^{(1-a_{i,j})} \\
&= \prod_{i < j} p_{i,j}^{a_{i,j}} (1 - p_{i,j})^{1-a_{i,j}}.
\end{aligned} \tag{1.26}$$

Notice that in equation (1.26) each entry of the adjacency matrix is treated as an independent and Bernoulli-distributed random variable, with different probabilities of success equal to

$$p_{i,j} = \frac{x_i x_j}{1 + x_i x_j} \quad \text{for } i \neq j$$

therefore the probability associated to the entire graph is simply given by the product of these marginals. The log-likelihood to maximise, in order to obtain an estimate for the vector of parameters \vec{x} , is given by

$$\begin{aligned}
\mathcal{L}(\vec{x}|\mathbf{G}^*) &= \ln P(\mathbf{G}^*|\vec{x}) \\
&= \sum_i k_i(\mathbf{G}^*) \ln(x_i) - \sum_{i < j} \ln(1 + x_i x_j)
\end{aligned} \tag{1.27}$$

and the result is the estimate \vec{x}^* that guarantees that the constraints on the expected degree sequence are satisfied, on average, in the ensemble. The same solution \vec{x}^* can be derived solving the set of coupled equations

$$\frac{\partial \mathcal{L}(\vec{x}|\mathbf{G}^*)}{\partial x_i} = \frac{k_i(\mathbf{G}^*)}{x_i} - \sum_{i < j} \frac{x_j}{1 + x_i x_j} = 0 \quad \forall i = 1, \dots, N$$

corresponding to the First Order Condition of the problem. Rearranging the terms, the expected degree of each node can be easily recovered as

$$k_i(\mathbf{G}^*) = \sum_{i \neq j} \frac{x_i^* x_j^*}{1 + x_i^* x_j^*} = \langle k_i \rangle_{\vec{\theta}^*} \quad \forall i = 1, \dots, N \tag{1.28}$$

and corresponds to the sum of the expected number of edges incident to node i . Indeed, the term

$$p_{i,j}^* = \frac{x_i^* x_j^*}{1 + x_i^* x_j^*} = \langle a_{i,j} \rangle_{\vec{\theta}^*} \quad \forall i \neq j$$

represents the probability that nodes i and j are connected in the graph. As in the previous example, equation (1.28) specifies equation (1.18) whenever the expected value of the degree of each node in the ensemble is fixed equal to its observed value. Also in this case, the notation $\langle \cdot \rangle_{\vec{\theta}^*}$ indicates the expected value of the quantity in parentheses under the probability distribution $P(\mathbf{G}|\vec{x}^*)$, being $x_i^* = \exp(-\theta_i^*)$ for all $i \in \mathcal{N}$ the maximum likelihood estimate of the Lagrangian multipliers vector.

1.3 The role of null models

The entrance of null models into the complex systems literature has been mostly driven by the requirement of explaining some behaviours and characteristics of observed real-world networks. For instance, the Barabási-Albert model model has been developed in order to shed light on the mechanism responsible for power-law distributed degree distributions. The reasoning behind the Exponential Random Graph framework goes in the same direction: given a real-world network of interest \mathbf{G}^* , a null model for it is constructed first introducing an ensemble of graphs with the same number of nodes as \mathbf{G}^* , then defining a probability distribution over the ensemble, that is determined maximizing the entropy of the system while fixing certain network quantities as constraints. Thus, the ensemble \mathcal{G} will preserve such properties on average, while keeping all the rest maximally random. Moreover, once the parameters of the probability distribution over \mathcal{G} are estimated with the maximum likelihood principle, the distribution is centered on \mathbf{G}^* and any other topological quantity can be unbiasedly estimated taking its average value on the ensemble. Let Y be a certain topological property that can be computed on \mathbf{G}^* . Using the same notation introduced in the previous paragraph, the

quantity

$$\langle Y \rangle_{\tilde{\theta}^*} = \sum_{\mathbf{G} \in \mathcal{G}} Y(\mathbf{G}) P(\mathbf{G} | \tilde{\theta}^*) \quad (1.29)$$

represents the *randomized value* of Y in the maximum-entropy graphs ensemble [35]. Thus, comparing $\langle Y \rangle_{\tilde{\theta}^*}$ with $Y(\mathbf{G}^*)$ (that is the value of the topological property Y computed on the real-world graph \mathbf{G}^*) it is possible to assess whether Y can be reconstructed starting from the enforced constraints only \tilde{X} or its value in the observed network requires additional information to be interpreted. However, the computation of the quantity in equation (1.29) strictly depends on the possibility to write equation (1.14) in a simpler form, without enumerating the entire set of graphs in the ensemble \mathcal{G} . This is possible only when considering constraints that are expressed in terms of the entries of the adjacency matrix of \mathbf{G} [35].

Once the development of a null model for \mathbf{G}^* is completed, the expected value of a certain network quantity $\langle Y \rangle_{\tilde{\theta}^*}$ in the ensemble can be obtained starting from its definition. For instance, the ensemble expected value of the average nearest neighbour degree $\langle k_i^{nn} \rangle_{\tilde{\theta}^*}$ can be obtained from equation (1.4). However, in order to facilitate this computation, a possible approximation of this quantity can be considered, simply substituting the expected values $p_{i,j}^* = \langle a_{i,j} \rangle_{\tilde{\theta}^*}$ obtained from the model into the expression of Y . For instance, for the cases of ANND or clustering coefficient ensemble average, it is sufficient to replace $a_{i,j}$ with $p_{i,j}^*$ into equations (1.3) and (1.4) to obtain an approximation of the expected quantities

$$\begin{aligned} \langle k_i^{nn} \rangle_{\tilde{\theta}^*} &\approx \frac{\sum_{j \neq i} \sum_{k \neq j} p_{i,j}^* p_{j,k}^*}{\sum_{j \neq i} p_{i,j}^*} \\ \langle C_i \rangle_{\tilde{\theta}^*} &\approx \frac{\sum_{j \neq i} \sum_{k \neq i,j} p_{i,j}^* p_{j,k}^* p_{i,k}^*}{\sum_{j \neq i} \sum_{k \neq i,j} p_{i,j}^* p_{i,k}^*}. \end{aligned} \quad (1.30)$$

This is the procedure adopted in [35, 36, 37, 133]. The comparison of these expected quantities with the values observed in the real-world network has a straightforward interpretation: whenever the average nearest-neighbour-degree (or the clustering coefficient) trend in the observed

graph resembles the average value in the ensemble, then we can conclude that the correlation (or clustering) patterns shown in the network can be substantially driven by the imposed constraints. On the contrary, in case the two trends show significant deviations, the assortativity and clustering patterns may be the result of a different underlying mechanism and the imposed constraints are not sufficient by themselves to reproduce these behaviours.

The presented ones are only two of the possible applications of the Exponential Random Graph framework. The randomization method can be easily adjusted to the cases of directed networks, specifying both the in- and out-degrees as a constraint [35], or weighted networks, either specifying the strength sequence only [35] or enhancing it with the degree sequence [37], or bipartite networks, specifying the degree sequence for both layers [36]. This last case will be briefly introduced in the following section, before discussing a method to recover monopartite networks starting from an initial bipartite structure.

1.3.1 Quantifying nodes' similarity in bipartite networks

In Section 1.1.2 of the Introduction we have proposed a simple example, in which the network of friendships within a group of people has been recovered starting from an initial bipartite structure, that takes into account the classes attended at school by each person in the group. This situation is an extremely simple example of *projection* of a bipartite network onto one of its layer. Indeed, starting from the initial bipartite structure that involves people and classes, we quantify the similarity between pairs of agents simply counting the number of classes they attend together at school. Using the same notation as in Section 1.1.2, this quantity is equal to

$$V_{i,j} = \sum_{\alpha \in \Gamma} m_{i,\alpha} m_{j,\alpha}$$

for all $i, j \in L$, where $V_{i,j}$ is the number of common neighbours (V-motifs hereafter) involving the pair (i, j) . See Figure 5 for a pictorial representation of these topological objects. A naïve projection method of a bipartite network onto one of its layers simply prescribes to connect a pair of

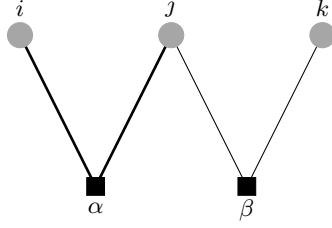


Figure 5: Graphical representation of two V-motifs: the first one $V_{\alpha}^{i,j}$ sees i and j that share α as a common neighbour on the opposite layer, while the second one, $V_{\beta}^{j,k}$, has j and k that are both neighbours of node β .

agents whenever they share at least one common neighbour in the bipartite structure. Thus the adjacency matrix of the final projected network will be given by $\mathbf{A} = \{a_{i,j} : i, j \in L\}$ where $a_{i,j} = \Theta(V_{i,j})$ and Θ is the Heaviside step function, equal to 1 whenever $V_{i,j} > 0$ and 0 otherwise. Therefore, in the friendship network, two agents are considered friends if they attend at least one class together.

As a matter of fact, projecting the information contained in a bipartite network on one of its layers is the traditional way of analysing many collaboration systems [19]. However, the naïve procedure described so far often generates a very dense projection, especially when the dimension of the graph increases. For this reason, many methods have been proposed in order to filter the final set of edges and evaluate the statistical significance of pairwise connections. In what follows we will review the proposal of [38].

Consider a bipartite network \mathbf{G}_{Bi}^* with biadjacency matrix \mathbf{M}^* . A benchmark model for this structure can be easily obtained extending the Exponential Random Graph framework introduced in Section 1.2.4 to the bipartite framework. In other words, the degree sequence involving nodes of both layers needs to be specified as constraint of the problem. Refer to [36] for a review of this methodology. Once the maximum-likelihood estimate of the Lagrange multipliers for the nodes of both layers (\vec{x}^*, \vec{y}^*) have been obtained, it is possible to recover the connection

probability for any pair of nodes in the system, as follows

$$p_{i,\alpha}^* = \frac{x_i^* y_\alpha^*}{1 + x_i^* y_\alpha^*} \quad \forall i \in L, \alpha \in \Gamma$$

and $p_{i,\alpha}^*$ is also the parameter of the Bernoulli-distributed random variable representing the edge $m_{i,\alpha}^*$ between nodes i and α . Therefore, denoting with $V_\alpha^{i,j} = m_{i,\alpha}^* m_{j,\alpha}^*$ the Bernoulli-distributed random variable indicating the presence of the common neighbour α between i and j , its distribution is simply given by

$$\begin{aligned} P(V_\alpha^{i,j} = 1) &= p_{i,\alpha}^* p_{j,\alpha}^* \\ P(V_\alpha^{i,j} = 0) &= 1 - p_{i,\alpha}^* p_{j,\alpha}^* \end{aligned}$$

for all $i, j \in L$ and $\alpha \in \Gamma$. Moreover, the previous term can be generalized to consider the number of common neighbours between i and j . Since all V-motifs are independent random variables with different probabilities of success, the number of common neighbours between i and j in the null model is given by

$$\langle V_{i,j} \rangle = \sum_{\alpha \in \Gamma} p_{i,\alpha}^* p_{j,\alpha}^*$$

for all $i, j \in L$, that is the parameter of the Poisson-Binomial random variable [39, 40]

$$V_{i,j}^* = \sum_{\alpha \in \Gamma} V_\alpha^{i,j} = \sum_{\alpha \in \Gamma} m_{i,\alpha}^* m_{j,\alpha}^*$$

indicating the observed number of common neighbours between i and j . At this point, the set of null hypotheses

$$\begin{cases} H_0 : & a_{i,j} = 0 \\ H_1 : & a_{i,j} = 1 \end{cases} \quad \text{for all } i, j$$

associated to each potential link in the projected network is tested. In other words, the significance of the *observed* number of motifs $V_{i,j}^*$ is tested through the computation of the following p-value

$$\text{p-value}(V_{i,j}^*) = \sum_{V_{i,j} \geq V_{i,j}^*} f_{PB}(V_{i,j})$$

where the term f_{PB} is used to indicate the Poisson-Binomial probability mass function [39, 40]. Repeating this procedure for all pairs of nodes in the graph, it is possible to obtain a squared matrix collecting the p-values associated to each potential link. Then, all p-values will be compared with a threshold value p_{th} in order to assess their significance. The numerical value of p_{th} is established through the False Discovery Rate procedure [41] that prescribes to

1. sort the vector of p-values in ascending order;
2. select the largest integer \hat{i} satisfying

$$\text{p-value}_{\hat{i}} \leq \frac{\hat{i}\xi}{N_L N_\Gamma} \quad (1.31)$$

where ξ is the chosen significance level;

3. consider $p_{th} = \text{p-value}_{\hat{i}}$ as the threshold value.

Then, all hypotheses which p-value is smaller than or equal to the threshold must be rejected, while we are not able to reject all hypotheses whose p-value is greater than p_{th} . In other words, the existence (i.e. significance) of each link is treated as a separate null hypothesis to test and only the edges associated to rejected hypotheses are included in the final projection. In the remainder, when using this method we will consider $\xi = 0.05$ as the significance value.

It is important to highlight that this validation procedure returns the validated network of the links associated to rejected hypotheses. Therefore, an edge in the final network indicates that the number of common neighbours between a pair of nodes was significantly higher than the model's expected value, under the assumption of independence of each link and under the considered set of topological constraints imposed in the null model's construction. In other words, the shape of the validated network strictly depends on the chosen benchmark model.

1.4 Outline of the work

The remainder of this work will be mainly focused on the analysis of the behaviour of the agents that are active in specific social network contexts. In this Section I will summarize the contents presented in each Chapter.

As highlighted in paragraph 1.3, the presented ones are only two of the possible applications of the Exponential Random Graph framework. Chapter 2 of this work provides a formal definition of rating networks specifying their topology and their interpretation in the real-world. Then the problems regarding the construction of a suitable null model for this kind of graphs is addressed, since the tools available for weighted networks turn out to be not appropriate in this context. Therefore, we extend the Exponential Random Graph framework for this specific type of networks, providing the theoretical background as well as some applications to real-world e-commerce datasets. The ability of the null model to reproduce some network properties (i.e. the neighbour connectivity and the number of Checkerboard motifs) is tested against some alternative models and the obtained results are discussed in great details. The comparison of observed properties with their ensemble averages is a tool available to understand to which extent the purchase activity of the users can be explained starting from their previous purchases. Indeed, with this model we obtain the probability that a users positively or negatively review a certain product, discounting for her/his previous purchase patterns.

Chapter 3 investigates how the users in the online social network of Twitter interact during the context of the electoral campaign that took place in Italy last March 2018. Using the projection and validation methods introduced in this Chapter, we have identified four groups of strongly connected verified users. We have also studied which of them turn out to be the most central in the validated network and studied how the remaining non-verified users interact with the four communities, highlighting a strongly polarized behaviour. Then, classifying the non-verified users according to their polarization values, we have studied which are the most used hashtags, noticing that the four communities tend to men-

tion their own coalition, followed by the major competitors. Finally, we have extended the validation procedure to bipartite directed networks: starting from the bipartite and directed network of tweets and retweets, we have projected and validated this structure in order to obtain a final directed network of retweets, in which a directed edge between two users exists if the second retweets the first one a significantly high number of times. We have studied how the information propagates in this graph identifying the most influential users and spreaders (i.e. the nodes with the highest out- and in-degrees respectively). The majority of the interactions still appear between users belonging to the same group. Finally, the analysis has been focused on the structure of the subgraphs made by the previous division in community. Interestingly, a sharp division is still present after the validation and some signals of internal split are also present, especially for the case of the right-leaning community.

Finally, Chapter 4 presents a stochastic model that reproduces the activities of a set of users in a social networks context. Given a system of agents, the work describes a model for the formation of the bipartite network of agents' actions and their features. This is a novelty in this literature, since we are only interested in describing the evolution of agents' activity and not in explaining how the connections among agents grow. Moreover, the choice of the features shown by each action is driven by a rule that we call *preferential attachment with weights*: the probability that a new action shows one of the features already present in the system does not only depend on the "popularity" of that feature (i.e. the number of previous actions showing it), but is also affected by some individual traits of the agents or the features themselves, synthesized in certain quantities called "weights". These weights can have different definitions and meanings according to the considered setting. We show some theoretical properties of the model and provide statistical tools for the parameters' estimation. Then, the model has been tested on three different datasets, two of them describe scientific collaborations in different research fields while the third one collects publications on a well-known online social network. The numerical results of the analyses are provided and discussed.

Chapter 2

Entropy-based randomization of rating networks

Carolina Becatti, Guido Caldarelli, and Fabio Saracco.

Physical Review E **99**, 022306 – Published 7 February 2019.

© 2019 American Physical Society

Abstract

In the last years, due to the great diffusion of e-commerce, online rating platforms quickly became a common tool for purchase recommendations. However, instruments for their analysis did not evolve at the same speed. Indeed, interesting information about users habits and taste can be recovered just considering the bipartite network of users and products, in which links represent products' purchases and have different weights due to the score assigned to the item in users' reviews. With respect to other weighted bipartite networks, in these systems we observe a maximum possible weight per link, that limits the variability of the outcomes. In the present article we propose an entropy-based randomization method for this type of networks (i.e. bipartite rating networks) by extending the Configuration Model framework: the randomized network satisfies the constraints of the degree per rating, i.e. the number of given ratings received by the specified product or assigned by the single user. We first show that such a null model is able to reproduce several non-trivial features of the real network better than other null models. Then, using our model as benchmark, we project the information contained in the real system on one of the layers; in order to provide an interpretation of the projected network, we run the Louvain community detection algorithm on the obtained graph and discuss the observed division in clusters. We are able to detect groups of music albums due to the consumers' taste or communities of movies due to their audience. Finally, we show that our method is also able to handle the special case of categorical bipartite network: we consider the bipartite categorical network of scientific journals recognized for the scientific qualification in Economics and Statistics. In the end, from the outcome of our method, the probability that each user appreciates every product can be easily recovered. Therefore, this information may be employed in future applications to implement a more detailed recommendation system that also takes into account information regarding the topology of the observed network.

2.1 Introduction

As mentioned in the previous Chapter 1, this part of the work deals with the problem of developing a suitable null model for *bipartite rating networks*. In this specific case, the two disjoint sets of nodes characterizing bipartite networks are represented by individuals and purchased items, while edges represent users' reviews and are weighted by the numerical score received by the product (e.g. see for example the well-known Amazon review system). Figure 6 provides a graphical representation of an example of rating network. Here, the three consumers on the upper layer purchase technological products on the lower layer from an e-commerce website. The edges connecting pairs of nodes identify reviews of products given by the three consumers and the stars associated to an edge represent the level of appreciation of the product.

These kind of graphs have been mostly studied from a machine learning and computer science perspective, in order to train models able to recommend items to people, based on their taste and preferences. Different methodologies are employed for this purpose, see [42] for a thorough review of the literature on the topic and the recorded progresses. In this work we focus on a different approach, providing an analytical tool that could reveal useful in the development of a recommendation system based on network topology. More specifically, we will adapt the Configuration Model framework, introduced in Chapter 1, to be suitable for this kind of graphs. In order to do so, we first need to identify the appropriate quantities that have to be included in the problem's specification as constraints.

Rating networks may be interpreted as classical weighted networks, which edges are weighted by a finite set of discrete scores. In this context, suitable constraints are represented by the specification of nodes' strengths only (as in the Weighted Configuration Model, in [35]). Because of the extremely poor predictive power of vertices' strengths, an enriched version of the previous model has been introduced (the Enhanced Configuration Model) in [43]; this method adds the topology as additional information. However, the presence (in our framework) of a finite number

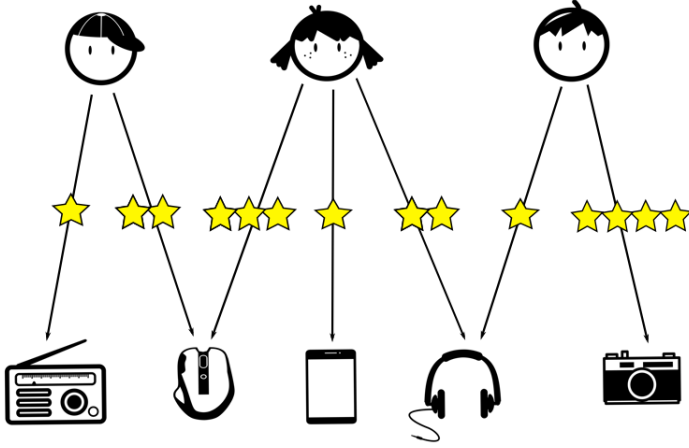


Figure 6: Graphical representation of a rating network. In this case three consumers buy technological products and assign them reviews with a score representing their level of appreciation of the item (e.g. the Amazon’s review system) [136].

of discrete weights complicates the problem formulation and increases the required computational effort. For these reasons, a preliminar “binarization” procedure is often employed (it is the approach of [38], but it is also common in recommendation systems, like in [44]), by thresholding the edges’ weights. In this way, the resulting network is binary and can be easily randomised with the Bipartite Configuration Model in [36].

In this Chapter we propose an alternative approach, constraining not only on the presence of positive reviews, but on the exact ratings distribution for each node. Due to its application, we indicate it in the following as Bipartite Score Configuration Model (BiSCM). The peculiarity of our approach is that we avoid the scores-related problems by specifying a *multi-degree* for each node in the network, i.e. by *specifying the entire distribution of scores received by a node*. We will show that the addition of more constraints allows to define a more restrictive null model, thus to reproduce with higher accuracy the features of the original net-

work. Let us highlight that our approach is general enough to allow to randomize bipartite signed networks (as a subset of rating networks) and bipartite categorical networks, the latter being a subject to which, to the best of our knowledge, there is a substantial scarcity of theoretical and analytical tools. For instance, while there are different proposals for measuring the similarity among items in categorical datasets [45], non-trivial null-models and benchmarks are practically absent. The present methodology tries to fill this gap.

The rest of this Chapter is structured as follows. In Section 2.2 we provide a formal definition of rating networks and explain the details of the entire ensemble construction procedure. In Section 2.3 we discuss potential limitations of the model, mainly due to its computational complexity, and describe some possible extensions to the cases of signed and categorical networks. In Section 2.4 we provide an outline of the methodology: we first define a possible generalization of the definition of some well-known topological quantities and see how our model is able to reproduce them; then we explain how our method can be introduced in the projection and validation procedure proposed in [38]. In Section 2.5 we describe the datasets used to develop the analysis, while in Section 2.6 we show the main results regarding the motifs and assortativity analyses. Moreover, we also describe the communities found in the projected and validated networks. Finally, we discuss possible future developments of the method in Section 2.7. The details regarding the development of the considered models are provided in Appendix A.

2.2 The model

In this section, we first introduce the required notation and then explain the necessary steps to construct the null model. As explained in previous Chapter 1, a *bipartite* network is a network which set of nodes can be partitioned in two subsets (i.e. layers), such that only edges between nodes belonging to different layers can be observed. The notation used in this chapter resembles the one introduced in Chapter 1: the symbols L , Γ and \mathcal{E}_{Bi} will be used to indicate respectively the sets of nodes in the

two layers and the set of edges in the network; the number of elements in the three previous sets will be denoted as N_L , N_Γ and E respectively. Finally, we will use $N = N_L + N_\Gamma$ for the total number of vertices in the graph.

A bipartite *rating* network can be entirely specified by its $N_L \times N_\Gamma$ biadjacency matrix $\mathbf{M} = \{m_{i,\alpha} : i \in L, \alpha \in \Gamma\}$ where the entry $m_{i,\alpha} = \beta$ indicates that product i has been reviewed and assigned score β by user α and $m_{i,\alpha} = 0$ otherwise. In what follows, we only deal with the case in which users are required to assign integer scores and the number of possible scores is known, denoted from now on as β_{max} , so that $\beta \in \{1, \dots, \beta_{max}\}$. Therefore, a null model for this kind of graphs is specified considering a benchmark ensemble of bipartite graphs \mathcal{G}_{BiS} , with constant number of vertices per layer, respectively equal to N_L and N_Γ . For the sake of simplicity, a binary representation of \mathbf{M} 's entries will be considered, defining $m_{i,\alpha,\beta} = \delta(m_{i,\alpha}, \beta)$ for all β , where δ is the classical Kronecker delta function. By doing so, we have

$$m_{i,\alpha,\beta} = \begin{cases} 1 & \text{if } m_{i,\alpha} = \beta \\ 0 & \text{otherwise.} \end{cases} \quad (2.1)$$

then the variable $m_{i,\alpha,\beta}$ will be equal to 1 if node α has reviewed node i with the numerical score β and $m_{i,\alpha,\beta} = 0$ otherwise. We use the notation

$$k_{i,\beta}(\mathbf{M}) = \sum_{\alpha} m_{i,\alpha,\beta} \quad i = 1, \dots, N_L \quad (2.2)$$

$$k_{\alpha,\beta}(\mathbf{M}) = \sum_i m_{i,\alpha,\beta} \quad \alpha = 1, \dots, N_\Gamma \quad (2.3)$$

to indicate the number of reviews with score β respectively assigned by a generic user α , in equation (2.2), and received by a generic product i , in equation (2.3). Notice that, as in the previous chapter, the terms $k_{i,\beta}(\mathbf{M})$ or $k_{\alpha,\beta}(\mathbf{M})$ are used to indicate, respectively, that the degree of nodes i and α are computed on the network represented by the biadjacency matrix \mathbf{M} . The specification of equations (2.2)-(2.3) for all $\beta = 1, \dots, \beta_{max}$ defines the distribution of scores received by each node and constitute the fundamental constraints of our problem.

At this point we look for the probability distribution maximising the (Shannon's) Entropy

$$S = - \sum_{\mathbf{M}} P(\mathbf{M}) \ln P(\mathbf{M}) \quad (2.4)$$

under the constraints on the expected degrees, i.e.

$$\begin{aligned} \langle k_{i,\beta} \rangle &= k_{i,\beta} \quad \text{for } i = 1, \dots, N_L \\ \langle k_{\alpha,\beta} \rangle &= k_{\alpha,\beta} \quad \text{for } \alpha = 1, \dots, N_\Gamma \end{aligned}$$

with $\beta = 1, \dots, \beta_{max}$. In other words, we consider the probability distribution over the ensemble such that the expected degree of each node, for every possible rating, equals on average an observed value while keeping all the rest maximally random. The solution to this bipartite maximization problem gives the following probability distribution over the ensemble

$$P(\mathbf{M}|\vec{x}, \vec{y}) = \prod_{i,\alpha} q_{i,\alpha}(m_{i,\alpha}|\vec{x}, \vec{y}) \quad (2.5)$$

where \vec{x} is a $N_L \beta_{max}$ dimensioned vector of Lagrangian multipliers that controls for the expected degree for each possible rating for users, while \vec{y} is the analogous $N_\Gamma \beta_{max}$ dimensioned vector of Lagrangian multipliers for the products. The quantity

$$q_{i,\alpha}(m_{i,\alpha}|\vec{x}, \vec{y}) = \frac{\prod_{\beta} (x_{i,\beta} y_{\alpha,\beta})^{\delta(m_{i,\alpha}, \beta)}}{1 + \sum_{\beta} x_{i,\beta} y_{\alpha,\beta}} \quad (2.6)$$

is the probability mass function of the discrete random variable $m_{i,\alpha}$ that takes values on the set $\{0, 1, \dots, \beta_{max}\}$. Refer to Appendix A for additional details. Notice that each node has been assigned a vectorial Lagrangian multiplier (\vec{x}_i if it belongs to the layer L , \vec{y}_α if it belongs to the layer Γ) of dimension β_{max} . Thus the probability to observe a single positive outcome, i.e. a link with rating β , can be expressed as

$$p_{i,\alpha,\beta} = \frac{x_{i,\beta} y_{\alpha,\beta}}{1 + \sum_{\beta} x_{i,\beta} y_{\alpha,\beta}} \quad (2.7)$$

for all i, α and β . Therefore, the outcome of our method allows to easily recover the probability that each user assigns a given score to all items, for all observed rating levels in the network.

In order to determine the numerical values for our Lagrangian multipliers, let us consider a specific real-world rating network \mathbf{M}^* , for which the degree sequences $\{k_{i,\beta}(\mathbf{M}^*) : i \in L^*, \beta = 1, \dots, \beta_{max}\}$ and $\{k_{\alpha,\beta}(\mathbf{M}^*) : \alpha \in \Gamma^*, \beta = 1, \dots, \beta_{max}\}$ are known for all nodes in the network. Again, the starred notation is used to indicate the sets of nodes in the observed network. Thus, at this stage, the graphs ensemble is built in such a way that there is a correspondence between the observed sets of nodes L^* and Γ^* and the sets of nodes in each graph of \mathcal{G}_{BiS} . The log-likelihood defined by equation (2.5) is given by

$$\begin{aligned} \mathcal{L}(\vec{x}, \vec{y} | \mathbf{M}^*) &= \sum_{i,\beta} k_{i,\beta}(\mathbf{M}^*) \ln x_{i,\beta} + \sum_{\alpha,\beta} k_{\alpha,\beta}(\mathbf{M}^*) \ln y_{\alpha,\beta} \\ &\quad - \sum_{i,\alpha} \ln \left(1 + \sum_{\beta} x_{i,\beta} y_{\alpha,\beta} \right) \end{aligned} \quad (2.8)$$

The maximization procedure consists in determining the specific parameters' values (\vec{x}^*, \vec{y}^*) that maximize the probability to observe the network of interest \mathbf{M}^* . The solution can be equivalently derived solving the following system of $N\beta_{max}$ coupled equations

$$\begin{aligned} \langle k_{i,\beta} \rangle &= \sum_{\alpha} \frac{x_{i,\beta} y_{\alpha,\beta}}{1 + \sum_{\beta} x_{i,\beta} y_{\alpha,\beta}} = k_{i,\beta}(\mathbf{M}^*) \\ \langle k_{\alpha,\beta} \rangle &= \sum_i \frac{x_{i,\beta} y_{\alpha,\beta}}{1 + \sum_{\beta} x_{i,\beta} y_{\alpha,\beta}} = k_{\alpha,\beta}(\mathbf{M}^*) \end{aligned} \quad (2.9)$$

for $i \in L^*$, $\alpha \in \Gamma^*$ and $\beta = 1, \dots, \beta_{max}$. Clearly, the system in (2.9) is obtained specifying the First Order Conditions for equation (2.8), i.e. setting its partial derivatives equal to zero. In other words, the equations in (2.9) underline that the solution of the maximum likelihood problem in equations (2.8) is exactly the vector (\vec{x}^*, \vec{y}^*) ensuring that the expected degree sequence coincides, for each score, with the observed one. For the applications presented in this work we have solved the system in equation (2.9). Once the null model construction is completed, the probability to observe a review with score β between nodes i and α can be easily recovered from the maximum likelihood estimates (\vec{x}^*, \vec{y}^*) , as follows

$$p_{i,\alpha,\beta}^* = \frac{x_{i,\beta}^* y_{\alpha,\beta}^*}{1 + \sum_{\beta} x_{i,\beta}^* y_{\alpha,\beta}^*}.$$

It is worth noting again that these values are calibrated on the data. Due to the properties of the maximum likelihood estimates, the vector (\vec{x}^*, \vec{y}^*) ensures that the expected degree of each node overall the ensemble, computed under the conditional probability distribution $P(\mathbf{M}|\vec{x}^*, \vec{y}^*)$, coincides with its observed value. This statements is translated into the following sequence of equalities

$$\begin{aligned}
\langle k_{i,\beta} \rangle_{\vec{\theta}^*} &= \sum_{\mathbf{M} \in \mathcal{G}_{\text{BiS}}} k_{i,\beta}(\mathbf{M}) P(\mathbf{M}|\vec{x}^*, \vec{y}^*) \\
&= \sum_{\alpha \in \Gamma^*} \sum_{\mathbf{M} \in \mathcal{G}_{\text{BiS}}} m_{i,\alpha} P(\mathbf{M}|\vec{x}^*, \vec{y}^*) \\
&= \sum_{\alpha \in \Gamma^*} p_{i,\alpha,\beta}^* = k_{i,\beta}(\mathbf{M}^*)
\end{aligned} \tag{2.10}$$

and analogously for nodes on the other layer. In equation (2.10) the term \mathbf{M} is used to indicate the biadjacency matrix of each graph in the ensemble \mathcal{G}_{BiS} . Equation (2.10) also implies that the randomized value of the degrees (or, in general, any network-related quantity that can be computed as a function of the biadjacency matrix entries) has a probability distribution over the ensemble that is centered on the observed value and therefore can be unbiasedly estimated.

2.3 Discussion on the model

In this Section we focus our attention on different aspects related to the null model's construction. In particular, obtaining the maximum likelihood solution of the model may become a hard task in terms of complexity, depending on the dimension of the considered graph and the total number of available rating levels. Therefore we here propose a possible alternative to approximate such a solution. Moreover, we also discuss some possible extensions of the same method to different categories of graphs.

2.3.1 Computational complexity

The null model's calibration (i.e. the determination of a numerical value for the Lagrange multipliers \vec{x} and \vec{y}) may easily become costly, since the number of unknowns of the problem grows linearly with the number of nodes in the network N and the number of observed scores β_{max} . As a rule of thumb, the time needed to obtain the link probabilities for a network of 525×25 nodes with a connectance of order 0.5 (therefore extremely high) is approximately three hours on a single process of a Intel(R) Xeon(R) CPU E5-2650 v2, 2.60 GHz server. However, this value should be considered as an upper bound for such a number of nodes, since the higher the connectance the higher the computation time (and 0.5 is a quite high connectance value). Regarding the fitnesses calculation (i.e. the Lagrangian multipliers), the complexity of the problem scales as $O(\beta_{max}(N_L + N_\Gamma))$.

For this reason, for extremely large and sparse systems, a possible approximation is here provided. As in the standard model presented in [46], for our Chung-Lu Approximation we define the fitness variables associated to each node to be proportional to nodes' degrees, as follows

$$x_{i,\beta}^{cla} = \frac{k_{i,\beta}}{\sqrt{E_\beta}} \quad \text{and} \quad x_{\alpha,\beta}^{cla} = \frac{k_{\alpha,\beta}}{\sqrt{E_\beta}}. \quad (2.11)$$

Then, to relax the constraints required, the connection probability between each pair of nodes in the network are obtained as

$$p_{i,\alpha,\beta}^{cla} = \begin{cases} x_{i,\beta}^{cla} x_{\alpha,\beta}^{cla} & \text{if } k_{i,\beta} k_{\alpha,\beta} \leq E_\beta \\ 1 & \text{otherwise} \end{cases} \quad (2.12)$$

for all $i = 1, \dots, N_L$, $\alpha = 1, \dots, N_\Gamma$ and $\beta = 1, \dots, \beta_{max}$. The term $E_\beta = \sum_{i,\alpha} m_{i,\alpha,\beta}$ in equations (2.11) and (2.12) identifies the total number of observed links for each score present in the data. Figure 7 provides a graphical comparison of the two definitions of probability for the ML network. It is evident that equation (2.12) systematically overestimates the BiSCM values, especially for high probabilities.

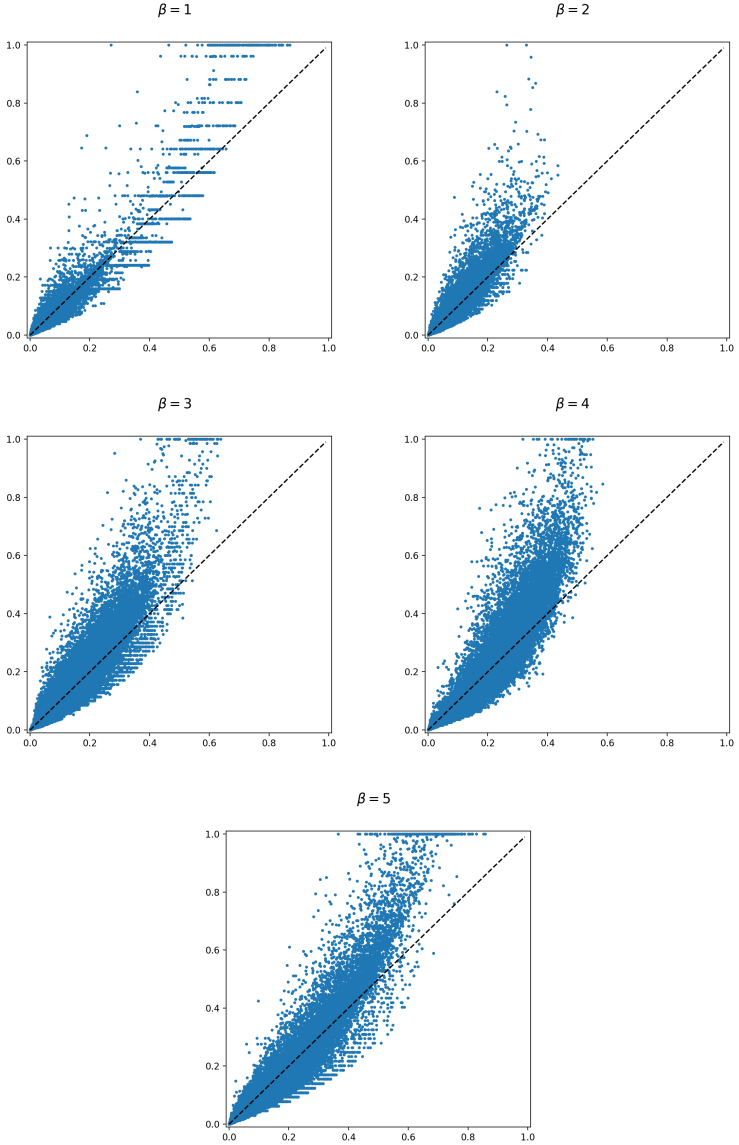


Figure 7: Graphical comparison of the two definitions of probabilities for one of the considered datasets, i.e. the MovieLens network [53, 138]. The plot shows $p_{i,\alpha,\beta}$ on the x -axis and $p_{i,\alpha,\beta}^{cla}$ on the y -axis.

2.3.2 Possible extensions

It is worth noting that no assumption has been made so far concerning the nature of the different entries of the adjacency matrix, but for the fact that they are mutually exclusive. Therefore, our method is completely general and can be employed in many different applications. For instance, this framework can be intended as describing a multiedge network in which β_{max} is the maximum number of edges allowed between any pair of nodes. Other possible extensions can be towards the cases of signed networks or categorical networks. In the former case, the different values of scores would be $\beta \in \{+1, -1\}$, indicating respectively presence of a positive and a negative link. In the latter case instead, each score β is simply assigned one of the possible realisations of a categorical variable: we will present an application to this kind of data in one of the following sections.

2.4 General methodology for the analysis

In order to test how accurately the proposed model reproduces the observed networks, we employ the following methodology: we first randomize the networks using the procedure explained in Section 2.2. Then, once the numerical estimates of the Lagrangian multipliers are obtained, we analytically calculate the ensemble expected value of some network-related quantities and we compare the observed values of such quantities with the model's average. Finally, again with the purpose of testing the accuracy of the obtained probabilities per link and per score, we employ a projection method to construct monopartite networks of products. Then we perform a community detection analysis and interpret the resulting groups. In the following Subsection 2.4.1 we provide a detailed description and interpretation of the topological quantities involved in the analysis, while Subsection 2.4.2 explains the necessary steps to perform the projection and validation procedures.

2.4.1 Higher order topological benchmark

Whenever dealing with numerical scores (let us exclude the case of categorical scores for the moment; we will discuss this situation later on), we are able to distinguish “positive” from “negative” reviews. More specifically, since in all cases β_{max} is known, we fix a threshold score to $\beta_{th} < \beta_{max}$ and we interpret each review as “positive” whenever the reviewed product has received a score greater than or equal to the threshold, meaning that it has been appreciated by the user. Clearly, the “negative” reviews are analogously defined as those that have been assigned a score smaller than the threshold, meaning that the user who assigned the review was not satisfied by the purchase. Therefore, at the end of the described randomisation procedure, we can separately consider the two categories of reviews defining a signed version of the original bi-adjacency matrix, indicated as $\overline{\mathbf{M}} = \{\overline{m}_{i,\alpha} : i \in L, \alpha \in \Gamma\}$, which entries can assume values $\overline{m}_{i,\alpha} = +1$ or $\overline{m}_{i,\alpha} = -1$ whenever a positive (i.e. $\beta_{th} \leq \beta \leq \beta_{max}$) or, respectively, negative (i.e. $1 \leq \beta < \beta_{th}$) review was registered in the observed data, and $\overline{m}_{i,\alpha} = 0$ otherwise. In what follows, the analysis has been performed using the binarisation $m_{i,\alpha}^+ = \delta(\overline{m}_{i,\alpha}, +1)$ and $m_{i,\alpha}^- = \delta(\overline{m}_{i,\alpha}, -1)$ where δ is again the Kronecker delta function. Clearly the shortcuts $m_{i,\alpha}^+$ or $m_{i,\alpha}^-$ indicate the presence of a score respectively greater or smaller than the threshold and can also be trivially defined in terms of the original data as

$$m_{i,\alpha}^+ = \begin{cases} 1 & \text{if } m_{i,\alpha} \geq \beta_{th} \\ 0 & \text{otherwise} \end{cases}$$

$$m_{i,\alpha}^- = \begin{cases} 1 & \text{if } 1 \leq m_{i,\alpha} < \beta_{th} \\ 0 & \text{otherwise} \end{cases}$$

Moreover, we denote the quantities

$$k_i^+(\overline{\mathbf{M}}) = \sum_{\alpha} m_{i,\alpha}^+$$

$$k_i^-(\overline{\mathbf{M}}) = \sum_{\alpha} m_{i,\alpha}^-$$

respectively *positive degree* and *negative degree*, to indicate the number of edges with positive or negative reviews incident to node i . The previous quantities are equivalently defined for the nodes on the opposite layer. Finally, the null models associated to positive and negative reviews can be easily obtained from the result of the previous randomization procedure. Indeed, the entries of the probability matrices associated to positive and negative reviews can be computed as

$$\begin{aligned}\langle m_{i,\alpha}^+ \rangle &= p_{i,\alpha}^+ = \sum_{\beta \geq \beta_{th}} p_{i,\alpha,\beta} \\ \langle m_{i,\alpha}^- \rangle &= p_{i,\alpha}^- = \sum_{\beta < \beta_{th}} p_{i,\alpha,\beta}\end{aligned}$$

where the terms $p_{i,\alpha,\beta}$ are defined in equation (2.7) and represent the probability that user i reviews product α with a numerical score equal to β .

On the available datasets, we first analyse the correlation between neighbour nodes' degrees introducing a signed version of the classical average nearest neighbor degree (ANND). We separately analyse all possible combinations of positive/negative neighbours and positive/negative degrees, as follows

$$\begin{aligned}k_i^{pp}(\overline{\mathbf{M}}) &= \frac{\sum_{\alpha} m_{i,\alpha}^+ k_{\alpha}^+}{k_i^+} \\ k_i^{pn}(\overline{\mathbf{M}}) &= \frac{\sum_{\alpha} m_{i,\alpha}^+ k_{\alpha}^-}{k_i^+} \\ k_i^{np}(\overline{\mathbf{M}}) &= \frac{\sum_{\alpha} m_{i,\alpha}^- k_{\alpha}^+}{k_i^-} \\ k_i^{nn}(\overline{\mathbf{M}}) &= \frac{\sum_{\alpha} m_{i,\alpha}^- k_{\alpha}^-}{k_i^-}.\end{aligned}\tag{2.13}$$

In the previous equations, the first apex letter is referred to the sign of the edges incident to i , while the second one indicates the sign of i 's neighbour degree. In other words, the terms k_i^{pp} or k_i^{pn} in the equations in (2.13) respectively identify the average positive or negative degree of

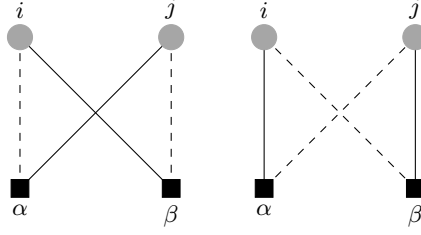


Figure 8: Checkerboard-like motifs. In this representation, continuous lines represent positive reviews while dashed lines represent the negative ones.

node i 's positive neighbours or, otherwise stated, the positive or negative degree of the other nodes that have assigned a positive review to i . Clearly, replacing positive with negative sign, we obtain the analogous interpretation for k_i^{np} and k_i^{nn} .

Then, we compute the number of signed checkerboard-like motifs c_i [47] for each node, as follows

$$c_i(\overline{\mathbf{M}}) = \sum_{\alpha, \beta} \sum_j m_{i, \beta}^+ m_{j, \alpha}^+ m_{i, \alpha}^- m_{j, \beta}^- \quad (2.14)$$

Equations (2.13) and (2.14) are analogously defined for column nodes α . The number of checkerboard-like motifs a node is involved represents the number of times we observe a pair of products that receives conflicting reviews from a pair of users. Considering the pictorial representation in Figure 8, we have that the products (i, j) receive opposite reviews from the two users (α, β) : in this case, user α has appreciated product j and instead disliked i , while for β we observe different preferences. Trivially, continuous edges represent positive reviews while dashed represent negative ones.

2.4.2 Monopartite projections

A possible way to analyse collaboration systems [19] (e.g. actors in the movie system), is to project the information contained in a bipartite network on one of the layers, then considering the statistical significance

of their common connections. From various attempts on boards [48] to more recent approaches [49], several works [38, 50, 51, 52] applied a similar idea, making use of the Bipartite Configuration Model. Summarizing, once the probability for the single bipartite link is calculated, it is possible to compute the probability that a pair of nodes shares a link with an item on the opposite layer. Such a pattern can be represented by a V-motif (see Figure 5). If the probabilities per link $\{p_{i,\alpha} : i \in L, \alpha \in \Gamma\}$ are independent, the probability to observe the single V-motif of Figure 5 is simply given by the product $P(V_\alpha^{i,j} = 1) = p_{i,\alpha} p_{j,\alpha}$. Thus, the number of common neighbours between i and j is a random variable with Poisson-Binomial distribution [39, 40], i.e. the distribution of N_Γ independent Bernoulli events, each with different (in general) probabilities. Comparing the observation on the real network with their theoretical Poisson Binomial distribution, it is possible to calculate a p-value for each pair of nodes on the same layer. After a multiple hypothesis testing procedure it is possible to state which connections of the monopartite projection are statistically relevant, therefore which are the nodes that share more connections than expected by the null model. See Section 1.3.1 of the previous Chapter for a more detailed description of this method.

In the present chapter, as an application of the BiSCM null-model as a benchmark, we shall extend such a procedure to rating networks, considering pairs of items that receive both positive reviews from the same customer. Indeed, the inclusion of extra constraints in the randomization phase, allows to construct a more restrictive null model. This fact is confirmed by the results of our validation procedure, since all validated networks are characterised by a very low connectance and the communities observed have a very precise interpretation (as shown in the following sections). If we set the threshold for positive reviews at β_{th} , the probability to simultaneously observe positive reviews for the items (i, j) from the same customer α is given by

$$P(V_\alpha^{i,j} = 1) = p_{i,\alpha}^+ p_{j,\alpha}^+. \quad (2.15)$$

Now, the algorithm follows exactly the same steps of the original procedure: from the probability that both items receive a positive review

from the same customer (2.15) we calculate the distribution of common good reviews. Although several methods are available in the literature, we employ the False Discovery Rate procedure [41] to validate the previously calculated p-values, since it permits to have a stricter control on the false positives. The entire procedure has been reviewed in Section 1.3.1 of the Introduction.

2.5 Datasets

The following datasets have been employed to test the proposed methodology.

- **MovieLens 100k (ML):** Bipartite network that collects 100,000 movies' ratings. The website's users are characterized by some individual features, such as age, job, sex, state and zipcode. For the set of movies we have information on the release year, title and genre. Each user can review a movie with a numerical score $\beta \in \{1, 2, 3, 4, 5\}$ according to her/his level of appreciation. We consider as positive all reviews assigned a score greater than or equal to $\beta_{th} = 3$. The data has been downloaded from the repository [138] while any additional information is provided in [53].
- **Amazon:** We collected three datasets involving different categories of products. From [139] we downloaded the Musical Instruments (MI) and Digital Music (DM) datasets, that respectively collect purchases of musical instruments and CDs or vinyls (the latter had to be further sampled due to its high dimensions). The data about Smartphones (SM) and related products has instead been downloaded from [140]. For all of them, the possible numerical ratings for each purchase are $\beta \in \{1, 2, 3, 4, 5\}$. As in the previous case, we consider as positive all reviews that receive a score greater than or equal to $\beta_{th} = 3$.
- **Anvur (AN)** is the acronym for "Associazione Nazionale di Valutazione del sistema Universitario e della Ricerca", an Italian agency

	N	N_L	N_T	E	d	+	-
ML	2588	1645	943	100000	$6.36 \cdot 10^{-2}$	0.83	0.17
MI	2329	900	1429	10261	$7.98 \cdot 10^{-3}$	0.95	0.05
SM	16172	2256	13916	15817	$5.04 \cdot 10^{-4}$	0.73	0.27
DM	7000	2500	4500	36774	$3.27 \cdot 10^{-3}$	0.91	0.09
AN	1396	15	1381	14903	$7.19 \cdot 10^{-1}$		

Table 1: Data description

that evaluates the quality of the Universities and research systems and determines which journals are considered top journals (and so the most influent) in any scientific area. For this application we have downloaded the free dataset [141] regarding the journals' classification in the scientific area of Economics and Statistics (Area 13). We then constructed the bipartite network of journals and scientific areas, in which a link exists if the journal is considered a top journal in the scientific area of interest. The two different scores are indicated in the original table as $\beta \in \{\text{green}, \text{red}\}$. The first one indicates that the journal is *currently* considered a top journal, the second one instead indicates that the journal was considered a top journal *only until December 2017*: all later publications will not receive the same classification.

A more detailed description of the datasets is provided in Table 1, where d denotes the connectance of the networks, while the symbols + and - indicate the percentage of positive and negative edges in each dataset (when available).

2.6 Results

For each dataset we employ the procedure described in Section 2.2 to construct the benchmark model. So we obtain a set of β_{max} probability matrices, one for every rating level, collecting the probability to observe the different ratings for each pair of nodes in the network.

Once the Lagrange multipliers' values (\vec{x}^*, \vec{y}^*) are obtained from the

likelihood maximisation problem in equation (2.8), the expected quantities $\langle k_i^{pp} \rangle$, $\langle k_i^{pn} \rangle$, $\langle k_i^{np} \rangle$, $\langle k_i^{nn} \rangle$ and $\langle c_i \rangle$ across the ensemble can be analytically computed starting from their formulation in equations (2.13)–(2.14). However for the following analysis, instead of the exact computation of these topological quantities, we have considered the approximation obtained simply replacing the terms $m_{i,\alpha}^+$ and $m_{i,\alpha}^-$ in equations (2.13)–(2.14) with their expected values $p_{i,\alpha}^+$ and $p_{i,\alpha}^-$. The same procedure has been followed in [35, 36]. Moreover, again following the instructions in [35] we have also identified a confidence region of two standard deviations around the average values. The comparison of observed and expected quantities indicates whether these higher order network properties can be explained by lower order topological structures, i.e. the constraints imposed on nodes’ degrees, or require further investigation since they represent an indication of some correlation patterns in the observed network. Figures 9–13 show the results of this comparative analysis on the MovieLens dataset. All the expected values have been computed averaging the values of k_i^{pp} , k_i^{pn} , k_i^{np} , k_i^{nn} and c_i over the number of nodes having the same degree in the network. In order to provide a reliable analysis of the proposed model, we compare the BiSCM performance with some alternative ones: weighted configuration model (WCM), partial bipartite score configuration model (PCM) and Erdős-Rényi random graph (RG). The main difference among their construction relies in the specification of the imposed constraints. However, we refer to Appendix A for a full description of these alternative models. Red lines show the average connectivity and number of checkerboards estimated by BiSCM (top-left panels). Magenta, blue and green lines represent instead the same quantities estimated by WCM, PCM and RG, respectively. In most of the cases, the overall data trend is well captured by our ensemble. For the case of k_i^{pp} , some observations remain outside the two standard deviations range, suggesting the possibility of extra correlations that cannot be directly traced back to the degree sequence alone, despite the full specification of scores’ distribution. The analysis of the other null models would lead to completely unreliable conclusions, since in most of the cases, the induced ANND baseline is not able to capture the data over-

all trend. This is true especially for the cases of WCM and RG. The best performing alternative null model is the Partial BiSCM, that relies on the same type of constraints upon which the BiSCM is based but imposed on a single layer.

Due to the evident difference on the percentage of positive and negative observed reviews, a different type of analysis has been performed on the remaining datasets, taking into consideration positive reviews only: the BiSCM outperforms even in this case, as the Appendix shows.

2.6.1 Monopartite Communities

For three of our datasets we have reported the results of the projection analysis. The ML and DM networks have been binarized and then projected on the products layer, i.e. the movies and musical products layers respectively. The AN dataset has instead been projected onto both layers, considering separately the two sets of edges representing “green” and “red” links. All projection algorithms require to connect a pair of nodes in the monopartite network whenever they share a common neighbor in the bipartite graph. However, our projected edges have been further validated using the procedure presented in [38] and explained in Section 2.4.2. In order to discuss the structure of the validated network, we apply the Louvain modularity-based community detection algorithm [54] that allows to detect clusters of nodes. However, this method is well known to be order-dependent [55]. Therefore, to overcome this limitation, we consider the outcome of several runs of the algorithm, obtained reshuffling the initial order of the nodes, following the recipe of [38]. Interestingly enough, this approach greatly increases the performance of the original algorithm and behaves similarly to the Combo algorithm [56]. In the following discussion we consider the partition in communities detected by the best performing algorithm between reshuffled Louvain and Combo.

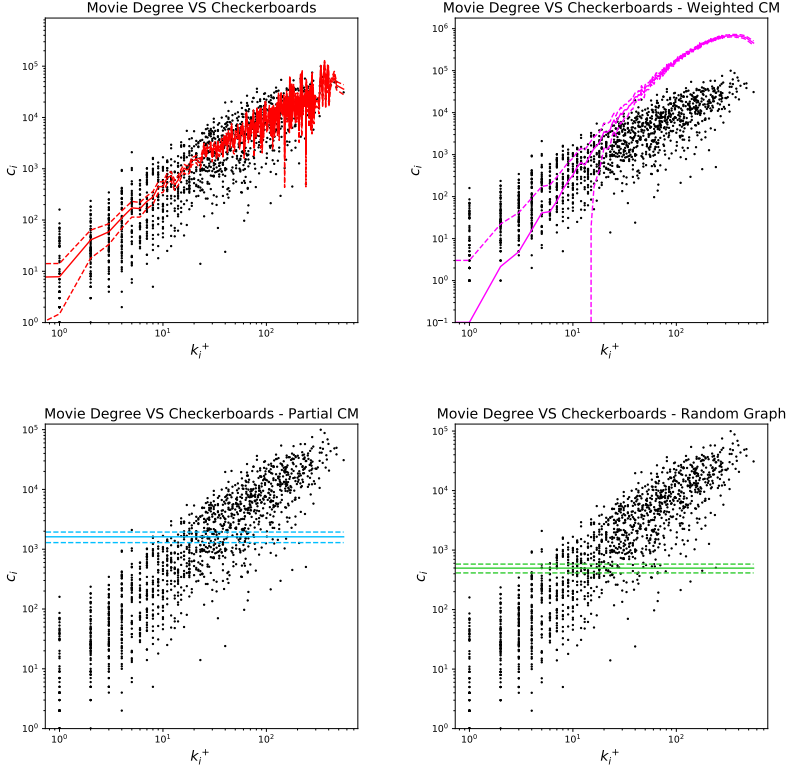


Figure 9: Application of the method to the ML network. The panels report c_i versus k_i^+ . The red line shows the expected values computed with our method. Magenta, blue and green lines are instead the expected values under WCM, PCM and Erdős-Rényi RG. The area of ± 2 standard deviations around the average value has been reproduced with dashed linestyle.

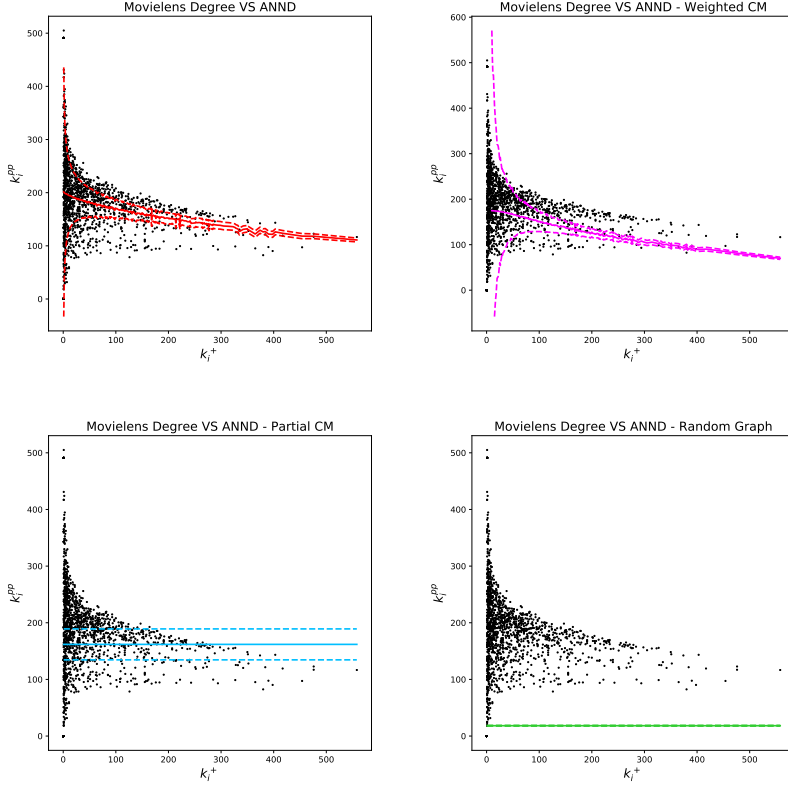


Figure 10: Application of the method to the ML network. The panels report k_i^{pp} versus k_i^+ . The red line shows the expected values computed with our method. Magenta, blue and green lines are instead the expected values under WCM, PCM and Erdős-Rényi RG. The area of ± 2 standard deviations around the average value has been reproduced with dashed linestyle.

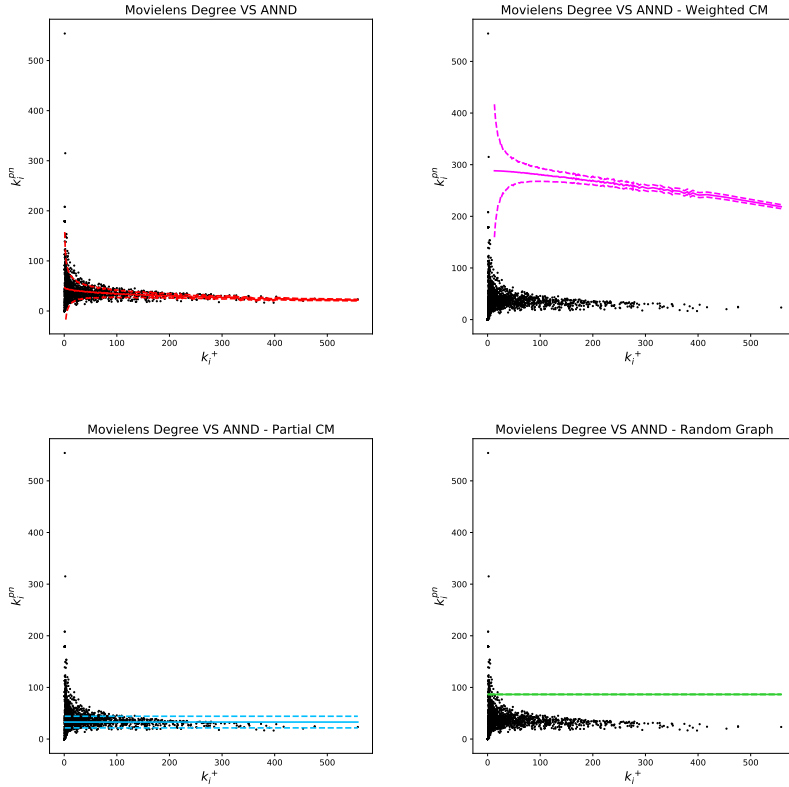


Figure 11: Application of the method to the ML network. The panels report k_i^{pn} versus k_i^+ . The red line shows the expected values computed with our method. Magenta, blue and green lines are instead the expected values under WCM, PCM and Erdős-Rényi RG. The area of ± 2 standard deviations around the average value has been reproduced with dashed linestyle.

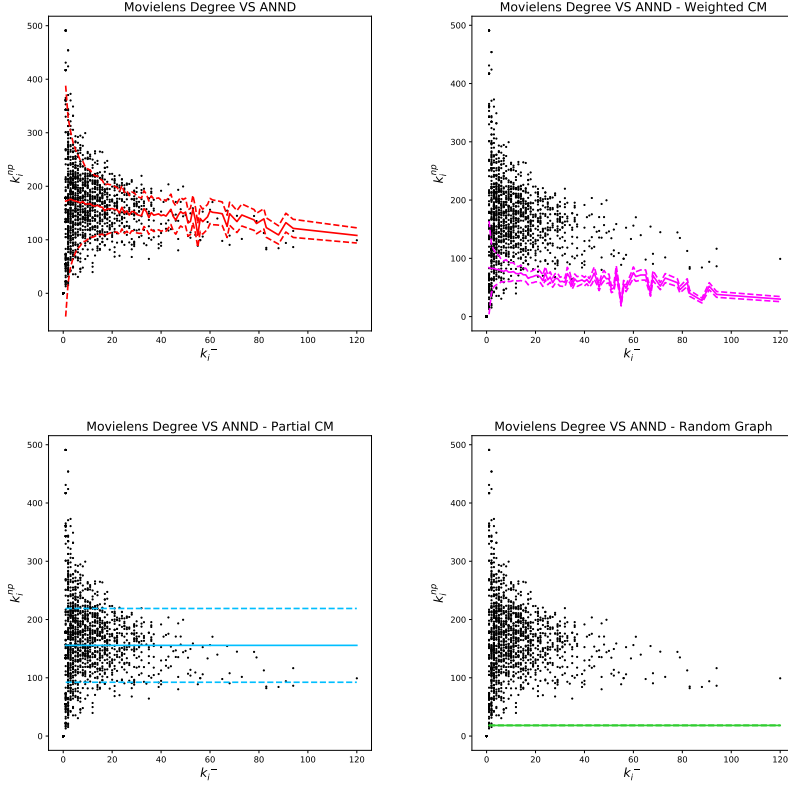


Figure 12: Application of the method to the ML network. The panels report k_i^{np} versus k_i^- . The red line shows the expected values computed with our method. Magenta, blue and green lines are instead the expected values under WCM, PCM and Erdős-Rényi RG. The area of ± 2 standard deviations around the average value has been reproduced with dashed linestyle.

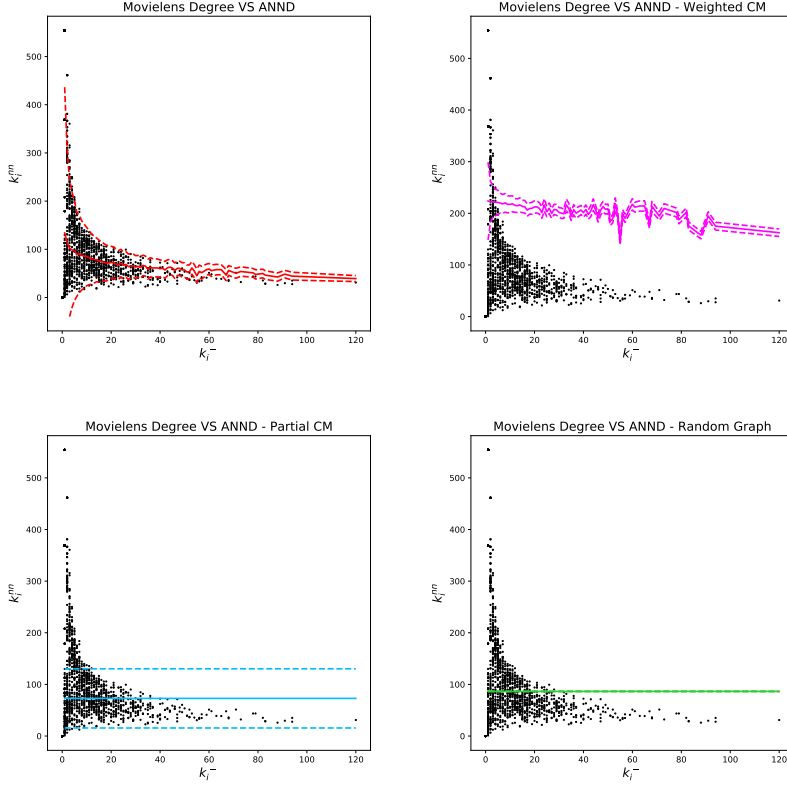


Figure 13: Application of the method to the ML network. The panels report k_i^{nn} versus k_i^- . The red line shows the expected values computed with our method. Magenta, blue and green lines are instead the expected values under WCM, PCM and Erdős-Rényi RG. The area of ± 2 standard deviations around the average value has been reproduced with dashed linestyle.

MovieLens

The partition in communities (obtained with the reshuffled Louvain algorithm, with a modularity of 0.451 and approximately 0.2% higher than the analogous Combo value) does not follow any genre-based division, as previously observed in [38], but rather identifies some characteristics shared by the movies' audience. The result of the community detection procedure is shown in Figure 14. Our method is able to detect movies released in 1996/97 the year before the survey (in orange), such as "Mission Impossible", "Independence Day", "Donnie Brasco". So this group of movies may be characterized by the curiosity of users towards new releases. A second group collects *family movies* (as they were called in [38]), including "Cinderella", "101 Dalmatians", "Home Alone" or "Mrs Doubtfire" (in green). In the blue community we find more "adult" movies, as the "Alien" saga, the episodes of "Die Hard", "Escape from New York", "Judge Dredd", "Conan the Barbarian", as well as "Terminator" episodes, and some westerns like "The Good, the Bad and the Ugly" and "Young guns". In this block we have also cult movies, such as "Blade Runner", "Star Wars", "Back to the Future". In the lime community we find horror titles, such as "Tales From the Crypt" episodes, "A Nightmare on Elm street" and "Bram Stoker's Dracula". The red community groups together European production movies ("Cinema Paradiso", "Mediterraneo", "Four Weddings and a Funeral", "Jean de Florette", "Como agua para chocolate"), but also US Independent production (as "Raising Arizona", "Clerks" or "Night on Earth"). Movies inspired by books or theatrical plays ("Emma", "Richard III", "Sense and Sensibility", "Othello") can be found in the pink community. In the last relevant block we find classical Hollywood movies (in yellow) such as "Casablanca", "Ben Hur", "Once upon a time in America", "Taxi Driver". Interestingly, this group also collects Hitchcock's filmography ("Vertigo", "Psycho", "Rebecca", "Rear Window"). Some smaller communities have however a very clear and defined characterization. For instance we have the community of all Wallace&Gromit short animation movies or the group of French production dramas (in magenta). Our

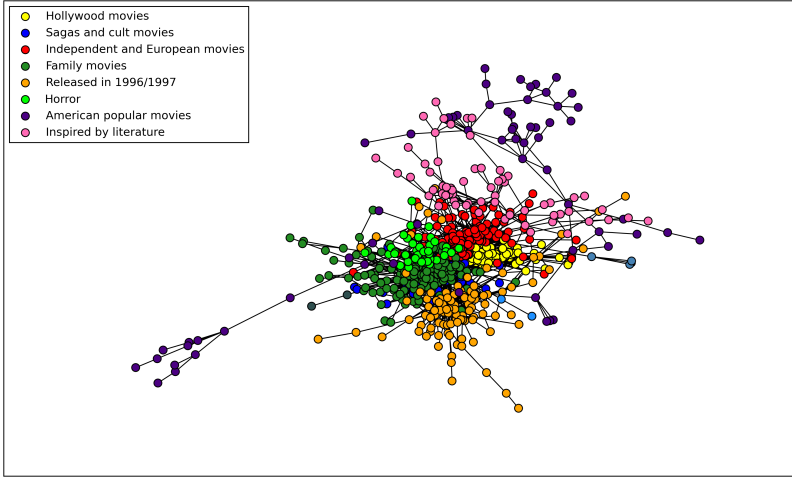


Figure 14: Graphical representation of the most numerous communities for ML networks. After the validation procedure, a standard modularity-based community detection algorithm is performed and the communities are here represented in different colors.

results are in substantial agreement with those of [38], but for the different connectance values of the BiCM- and the BiSCM-induced projection networks. Such behaviour is not surprising, because of the different constraints imposed by the two models. Indeed the BiSCM fixes the degree sequence for each rating in the bipartite network, while the BiCM only enforces the degree sequence of positive ratings (i.e. merging the information of 3, 4 and 5 stars). Therefore, while the former is more restrictive, the latter allows for greater fluctuations. This effect can be observed in the connectance of the validated projections, that is 0.87% for BiSCM against a value of 1.17% for the BiCM.

Digital Music

With respect to the previous ML case, here we obtain smaller and more precise groups of artists. In this case, the communities have been detected via the Combo algorithm and the final configuration has a mod-

ularity 0.874, approximately 0.001% higher than the reshuffled Louvain best partition value. For the DM network, each community reveals a specific genre or combination of genres. A pictorial representation of the most numerous communities of the validated network is provided in Figure 15.

We have the small light green community with two classic rock English bands, both of them characterised by a fusion with classical arrangements (Moody Blues and Electric Light Orchestra). Different clusters collect different shades of rock: the hard rock/heavy metal community is in blue (Loverboy, Alice Cooper, Van Halen, Scorpions, Deep Purple, Lynyrd Skynyrd) while the progressive rock is in green (Premiata Forneria Marconi, Soft Machine) and experimental rock in magenta. In dark violet we find the grunge rock and related similar tendencies, such as Alice in Chains, Pearl Jam, Soundgarden, as well as Red Hot Chilli Peppers, Iggy and the Stooges and the MC5. In indian red there is a community including Elton John, Billy Joel, the Genesis as well as Phil Collins and Peter Gabriel in their solo career. The sea green and indigo groups represent respectively female R&B singers (Whitney Houston, Aretha Franklin, Alicia Keys, Nelly Furtado) and female folk/pop ones (Alanis Morissette, Anastacia, Vanessa Carlton, Dido). The rap genre is divided between east coast and west coast hip hop, gangsta rap and a mixed community with the most famous artists (Eminem, Jay Z, 2 Pac, 50 Cent, D12), respectively depicted in light blue, dark magenta, lime and yellow. The isolated community in violet collects jazz contributors (Thelonious Monk, Miles Davis, Cannonball Adderley, Charles Mingus, Sonny Rollins), while the red one contains almost exclusively James Brown albums. A folk/country community (almost exclusively composed by John Denver and Gordon Lightfoot albums) is represented in gold. We finally have the grey and white groups with indie rock artists (Radiohead, Bon Iver, Of Monsters and Men) and the R&B singers and songwriters in pink (Marvin Gaye, Johnny Gill, Luther Vandross). In orange there is the community of folk/rock/ blues, including Bob Dylan, Jimi Hendrix, Eric Clapton, the Who, Paul Simon but also the subsequent Elvis Costello and Bruce Springsteen. It is interesting to find here even

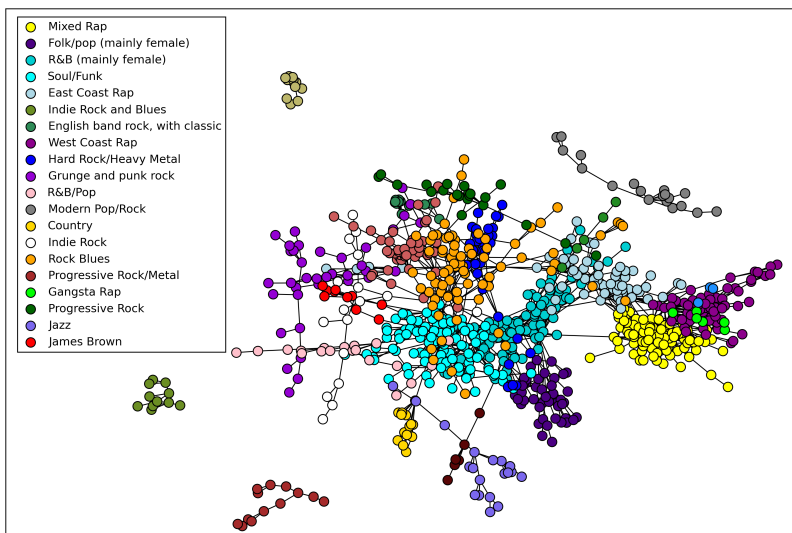


Figure 15: Graphical representation of the most numerous communities for DM networks. After the validation procedure, a standard modularity-based community detection algorithm is performed and the communities are here represented in different colors.

Robert Johnson, the legendary bluesman, who was a source of inspiration for the artists in this community. The community of soul-funk (the Jackson 5, Barry White, Stevie Wonder, the Commodores, the Parliament, Sly and the Family Stone, Prince, the Isley Brothers) is in cyan. It is interesting to note that some Jamiroquai albums have been detected in this latter community, since several experts compared the first production of this artist to Stevie Wonder [142]. Some smaller communities have not been included in the plot due to their low number of participants. However their interpretation is still clear, since they generally collect single artists (Leonard Cohen) or identify a very specific music genre (such as the group of white rappers Insane Clown Posse and Anybody Killa of genre horrorcore).

The case of categorical scores

As previously highlighted, in this section we present one of the possible extensions of the methodology. More specifically, we show an application to the case in which the distinct scores represent different realizations of a nominal variable. In this case, we use the Anvur dataset presented in the Section 2.5, where the categorical scores represent the classification of journals in the considered scientific areas, i.e. currently top journals or top journals only until December 2017.

First of all, we construct the benchmark model following the procedure explained in Section 2.2. By doing so, we obtain two probability matrices, one for each β , representing the probability that each journal is currently considered a top journal in each scientific sector or it is considered a top journal only until December 2017. Then, the analysis proceeds differently with respect to the previous cases. Indeed, due to the impossibility to fix a threshold beta given the nature of the analysed scores, we validate the projected networks using the two probability matrices as separate benchmarks. Through the validation on the scientific sectors layer we are able to understand which groups of subjects share a significant number of common neighbours, i.e. they both consider the same journals as relevant for the topic of interest. The validation on the other layer instead identifies the groups of journals that are considered top journals (or not) for the same scientific area in the Economics and Statistics field.

Figure 16 shows the communities identified after the validation on the scientific sectors layer, using the benchmark model obtained with $\beta = \text{green}$. However, we got the same result with the other benchmark model, with $\beta = \text{red}$. Clearly the three disconnected communities represent the division in different research topics: the sea green community identifies the two macro areas 13/A and 13/C, respectively corresponding to Economics and Economic History; the other two pink and orange groups instead represent the sectors 13/B and 13/D, respectively corresponding to the research fields of Business Administration and Statistics. Conversely, the validation on the other layer does not recognize any sig-

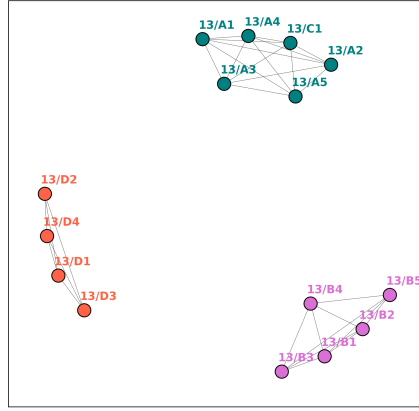


Figure 16: Graphical representation of the communities identified in the An-vur dataset. The validation procedure has been performed on the scientific sectors layer.

nificant link between pairs of journals, with both the considered benchmarks. This result is not so surprising, since a similar behaviour was previously observed when projecting the World Trade Web [38, 52]: when the rectangularity of the bipartite network, i.e. the ratio between the two dimensions of the layers, is particularly high, the most numerous layer has a lower variability due to the smaller dimension of the support of the Poisson-Binomial distribution. Moreover, since the effective significance threshold of FDR is even reduced by the total number of nodes' pairs on the layer, the possibility of finding a significant p-value is further reduced.

2.7 Discussion and conclusions

In everyday web experience we encounter many different examples of online review platforms: from Amazon customer ratings, to Tripadvisor

or Anobii, just to mention the most renowned ones. All these services provide an incredible source of information: indeed they are currently used to train recommendation systems, in order to show possible items' commercials that resemble customers' taste [44, 57, 58, 60, 61, 62]. Nevertheless, to the best of our knowledge, a proper method to use as a benchmark model for this kind of systems is not available in the literature.

In order to fill this gap, we follow the research line of entropy-based null models [33, 34, 35] that provides an unbiased framework. In the case of score networks, the main difficulty resides in considering mutually exclusive outputs for each entry of the biadjacency matrix, i.e. having different possible scores with different probabilities. Our approach resembles the one presented in [37] for the reciprocal configuration model. In that case, four mutually exclusive possibilities were observable for every pair of nodes: no link, an exclusively outgoing link, an exclusively ingoing link or a reciprocal link. Following this track, we were able to extend the Configuration Model framework to rating networks. We have shown that our method can be applied to many other types of networks, such as signed networks or categorical bipartite network.

Once the randomization procedure is complete, many types of analysis are possible. We first show that the obtained benchmark ensemble is able to capture some non trivial network information, like the abundance of topological patterns such as the extensions of the ANND and bipartite motifs to rating networks. Then, another application was proposed: the model was employed as the benchmark for the validation procedure presented in [38], in order to filter the information contained in one of the two layers. Otherwise stated, if the number of edges' co-occurrences is significantly higher than the null model expected value, then a link between the involved nodes is validated. In this sense, we compare the real network with the expected value of the BiSCM randomization: the disagreements are signals of non-trivial similarities among nodes on the same layer and therefore a link between them is included in the projection. The result of such a procedure is a monopartite undirected network describing nodes belonging to the same layer performing similarly in the bipartite system. In order to have a clear understanding of the structure

of the projected network, we run the Louvain community detection algorithm on it. Analysing the Amazon Digital Music dataset [139] we were able to uncover communities of music albums based on customers taste. Analysing the dataset of [138] it was instead possible to refine the community detection of [38]: indeed our model is more constrained than the one proposed there and introduces more filters than the simple BiCM after the binarization. Finally, we provide an example of possible application to the case of categorical networks. In this case, the standard division of research areas has been recovered just observing the number of top journals they share.

Therefore, the knowledge of the success probability, for every score level, for every pair of nodes in the system, may provide significant insights regarding customers' purchase habits and preferences. Moreover, our methodology recovers these values unbiasedly and with no a-priori information about the involved users, but simply observing the topology of the network itself. Therefore, it may be suitable for different future applications. As an example, in the last years entropy-based configuration models [33, 34, 35] were successfully used for link prediction [59]. Indeed, a similar approach can be addressed to tackle the problem of predicting links in a review context using BiSCM, taking advantage of the general properties of Configuration Models. We leave this analysis for future developments of the research.

Chapter 3

Extracting significant signals of news consumption from social networks: the case of Twitter in the Italian political elections

This Chapter is a joint work with my supervisors Prof. Guido Caldarelli and Dr. Fabio Saracco, and Prof. Renaud Lambiotte, my supervisor during the three months I spent as a visiting student at the University of Oxford. The full text of the article is also available from the arXiv repository, preprint number 1901.07933.

Abstract

According to the Eurobarometer report about EU media use of May 2018, the number of European citizens who consult on-line social networks for accessing information is considerably increasing. In this work we analyze approximately 10^6 tweets exchanged during the last Italian elections. By using an entropy-based null model discounting the activity of users, we first identify potential political alliances within the group of verified accounts: if two verified users are retweeted more than expected by the non-verified ones, they are likely to be related. Then, we derive the users' affiliation to a coalition measuring the polarization of unverified accounts. Finally, we study the bipartite directed representation of the tweets and retweets network, in which tweets and users are collected on the two layers. Users with the highest out-degree identify the most popular ones, whereas highest out-degree posts are the most "viral". We identify significant content spreaders by statistically validating the connections that cannot be explained by users' tweeting activity and posts' virality by using an entropy-based null model as benchmark. The analysis of the directed network of validated retweets reveals signals of the alliances formed after the elections, highlighting commonalities of interests before the event of the national elections.

3.1 Introduction

The phenomenon of diffusion in disordered media is a standard problem in Statistical Physics [64]: it has been described following different approaches, see for example the studies dealing with invasion [65] or percolation [66]. Moreover, in the recent years, the interconnection between individuals in the society formed a different non-regular lattice on which news and gossip propagate from one person to another one [67]. Indeed, the way Europeans have approached news and information has drastically changed: according to the last Eurobarometer report about EU media use [68], printed press is consulted everyday by 28% of the EU citizens, following a decreasing trend, while it is never consulted by approximately 20% of them. On the other hand, the daily access to the Internet increased up to the 65% of the population (starting from a 45% in 2010) and the increase in the percentage of citizens using everyday on-line social network for accessing information is even more striking, going from 18% in 2010 to 42% in 2017. This movement towards new technologies, beside being different from country to country (for instance, in Belgium the everyday access to the web concerns the 72% of the population against the 53% of Italy), is paradoxically in opposition to a general distrust towards new media: only 20% of the population declared to trust the contents available in on-line social networks, 34% trusts the Internet in general, while radio (59%) and TV (51%) are considered more reliable.

In order to provide a global understanding of the structure and dynamic of these new media that are directly affecting our daily life, studies based on complex networks [8, 109] tackled different aspects of these phenomena. Different works analyzed the raising of grass-root democratic protests as Arab springs [69], Occupy Wall Street [70] or the Spanish “Indignados” [71]. Analogously, the dynamics of the electoral campaign online media has a relatively long literature, focusing, from time to time, on USA [72, 73, 74, 75, 76, 77, 78], Australia [79, 80], Norway [81], Spain [82], Italy [83, 84, 85], France [86] and UK [87, 88]. In general, the shift from mediated to disintermediated news consumption has led to a range of documented phenomena: users tend to focus on informa-

tion reinforcing their opinion (this is the phenomenon of *confirmation bias* [89, 90, 91, 92, 93]) and to group in clusters of people with similar viewpoints, forming the so called *echo chambers* [90, 91, 92, 93, 94, 95]. The different dynamics that the public debate follows on social-network platforms is also remarkable: the time evolution of viral non-verified contents is more persistent than the verified equivalent [91] and “negative” messages spread faster than “positive” ones, even if the latter reach on average a wider audience [96]. Moreover, the analysis of time evolution of the activities in social platforms helps to predict the trend of retweets [97], the interactions of a single user with her/his neighbours [98] and to detect future developments of information campaign at an early stage [99] or “astroturf” campaigns [86].

Extracting information from on-line social networks is often complicated by their complex, intertwined organization and their strong heterogeneity. Thus, salient features can be identified as significant deviations from carefully-constructed null models. The instructions given in Chapter 1, Subsection 1.2.4, provide a general framework for the analysis of real-world networks [105, 106], since they allow to build an unbiased entropy-based null-model following the information theory derivation of statistical physics [100]. It is probably worth stressing again the crucial role played by the constraints in the above framework: in order to provide a reliable benchmark, the constraints should represent an important property of the system under analysis. Depending on the application, they may represent either the total number of links, as in the Erdős-Rényi null model, or the degree sequence [33, 34, 35] (both these cases have been carefully described in Subsection 1.2.4) or other topological properties for the real network of interest [37, 101, 102, 103, 104, 133].

In the present work, we analyze approximately 10^6 tweets exchanged during the last Italian elections held in March 2018. Using entropy-based null-models, we are able to detect non-trivial patterns in the diffusion of viral contents and different diffusion strategies depending on the polarization of users. We employ a first undirected representation of the network of retweets by distinguishing between certified and non-certified users. The former group is predominantly composed of celebrities and

official accounts, including politicians, newspapers, etc. We then derive the single user affiliations to a political alliance, first identifying groups of verified users by their interaction with the opposite layer, following the recipe of [38, 50]. Indeed, if two verified users are retweeted more than expected by non-verified ones, they are likely to be related. In other words, we exploit the way normal accounts consume the news on the social networks, in order to detect groups of strongly connected certified users. We analyse the community organisation of the resulting network and measure the polarization of unverified users: as observed in other studies [89, 90, 91, 92, 93, 94, 95, 107, 108], people tend to interact just with a single community, strongly polarizing their opinions. Indeed, we confirm this observation in the Italian elections of 2018. Finally, we study a bipartite and directed representation of the users' tweets and retweets network, in order to identify significant news consumers. In this context, we detect both the most popular and most significant content spreaders for the different communities. In the former case, we focus only on the users with the highest out-degree, in the latter case we statistically validate the directed connections that cannot be simply explained by the "virality" of the tweets and the tweet/retweet activity of the users. We indeed highlight different aspects of the diffusion of information on the network of users. This last validation is an extension of the approach of [38, 50] to direct bipartite networks.

The rest of the Chapter is structured as follows: in Section 3.2 we first introduce the dataset, describing how the data has been gathered and some of the available information; then we describe the employed network-based representations, as well as the necessary methods to analyse the considered networks in Section 3.3. The outcomes of the different analyses are described in Section 3.4, whereas we summarize research questions, results and this work's contribution in the final Section 3.5. Refer to Appendix B for a detailed description of the ensemble construction method.

3.2 Dataset

By means of the Twitter API, we have downloaded a sample of all tweets posted on Twitter from January 28 to March 19, 2018. The query has been performed with the only requirement that each post must contain at least one of a set of elections-related keywords in Italian language, such as “elezioni”, “elezioni2018”, “4marzo”, “4marzo2018”, meaning “elections”, “elections2018”, “4march”, “4march2018” respectively. This process returned approximately 1 million tweets.

Each of them provides several information regarding the tweet or its author, such as: a unique tweet ID; the date and time in which the post was created; the first 140 characters of the text; the unique identifier of the user who published the post, as well as her/his screen name, account name and the description that appears in her/his profile page; a Boolean variable indicating whether the user’s account has been certified as authentic; the geographical coordinates of the place where the post was published, when available due to users’ privacy restrictions; the number of replies and retweets the post has received; a list of entities contained in the post, such as hashtags, url addresses, users mentions, symbols, media; the language used in the text; a Boolean variable indicating whether the post contains possibly sensitive content; a Boolean variable indicating whether the tweet is actually a retweet of a previous post. Notice that, in case we are dealing with a retweet, the same material is available for the original tweet as well. Moreover, due to the structure of the data provided by the API, it is not possible to identify chains of sequential retweets. For instance, if A posts something, then B retweets from A and C retweets from B’s retweet, the data simply indicates that both B and C have retweeted some content from the initial user A.

3.3 General methodology for the analysis

The outline of our analysis is split into three major steps: in the first part we focus on the identification of some groups of verified accounts: using tools borrowed from network analysis we identify four communities

of strongly connected accounts and we consider them as “labels” for the verified users included in the data. Then, we aim at extending the assignment of these categories also to the group of non-verified accounts. In order to do so, we propose a possible definition of polarization index and we compute it for all the non-certified users, in order to determine their affiliation. Finally, we propose a novel representation of the tweeting and retweeting activity as a bipartite and directed network. Using this framework, we are able to construct a suitable null model for this graph and employ it to detect the most active users in terms of interactions with the others. Each of the previous steps will be further explained in the following Subsections.

3.3.1 Classification of verified users

In this paragraph we describe the methods employed in order to identify groups of strongly connected verified accounts. Once detected, we try to interpret those communities and we consider them as “labels” to classify the involved users.

Bipartite network of verified and non-verified users

As a first step, we have split the sample of available and Italian-speaking users in two categories, the groups of *verified* and *non-verified* users. Each account can request to be verified by the system: by doing so, Twitter guarantees that the account is authentic. This kind of procedure is, in general, applied to all those people who are considered of public interest. Therefore, we expect the accounts of famous people, politicians, newspapers, TV channels, radio channels etc. to be included into the set of verified users, while all the remaining users to belong to the other set.

Using the notation introduced in Chapter 1, we denote with $T = \{\text{Jan28}, \dots, \text{Mar19}\}$ the set of days in our observation period and we indicate with L_t and Γ_t respectively the sets

$$\begin{aligned} L_t &= \{i : i \text{ is an active, verified user during day } t\} \\ \Gamma_t &= \{\alpha : \alpha \text{ is an active, non-verified user during day } t\} \end{aligned}$$

for $t \in T$. Therefore L_t and Γ_t represent the sets of unique users, respectively verified and non-verified, who posted some contents (tweets and/or retweets) on Twitter during day t . Given this division, we construct a first bipartite network from the data, the network of retweets between verified and non-verified users during the whole period $\mathbf{G}_{\text{Bi}}^* = (L, \Gamma, \mathcal{E}_{\text{Bi}})$, with

$$L = \bigcup_{t \in T} L_t, \quad \Gamma = \bigcup_{t \in T} \Gamma_t$$

and $\mathcal{E}_{\text{Bi}} \subseteq L \times \Gamma$. Thus, the network is built with the sets of verified and non-verified users who were active during the considered time period.

By definition of bipartite networks, with this representation we exclude all the cases in which two certified (or non-certified) users retweet each other. We do so since we are interested in exploiting the way normal people consume the news to detect connected groups of certified users. Moreover, notice that in this case the edge $(i, \alpha) \in \mathcal{E}_{\text{Bi}}$ indicates that either i or α has retweeted the other's content *at least once* during the considered time period, we do not take into account the specific number of retweets observed between a couple of users. Notice also that at this step we do not consider the direction of the edges, therefore in principle we do not know who is the author of the post and who is the second one who shares the content. However, the activity of the verified users (especially political figures) is often limited to posting content on the social media, while the actions of retweeting, replying and sharing are mostly due to their audience, i.e. non-verified accounts, as shown in Figure 17 for the cases of tweets and retweets.

We denote with $\mathbf{M}^* = \{m_{i,\alpha}^* : i \in L \text{ and } \alpha \in \Gamma\}$ the biadjacency matrix of the real network \mathbf{G}_{Bi}^* where its entries $m_{i,\alpha}^*$ are equal to 1 if there exists at least one retweet between nodes i and α and 0 otherwise. The final number of nodes and edges in the network is specified in Table 2. As introduced in Section 3.1, this bipartite network of retweets between verified and non-verified users is used to detect political alliances among the verified accounts, as well as to identify the affiliation of non-verified users towards such a division.

	N_L	N_Γ	E	$\langle k \rangle$	k_{min}	k_{max}	d
Retweets	866	44058	102508	4.56	1	5862	0.003

Table 2: Some quantities in the network of retweets between verified and non-verified users: number of nodes per layer, total number of edges (i.e. retweets), average degree, minimum and maximum value of the observed degrees, connectance of the network.

Identifying connected groups of certified users

In order to understand whether the audience is polarized towards the source of information that better resembles their ideology, we first need to identify groups of connected users that (potentially) represent the current Italian political scenario. With this purpose in mind, we use the bipartite network \mathbf{G}_{Bi}^* described in the previous paragraph and we first build a null model for this observed graph using the methodology explained in Section 1.2.4 of Chapter 1. Essentially, we build a null model for this kind of networks using the degree sequence of both layers as constraints of the problem. Refer to [36] for a detailed summary of the method.

Once the bipartite network of interactions is built, we perform a projection procedure on the verified users layer. In other words, we construct a new monopartite network of verified users in which an edge between two users exists whenever they have interacted at least once with the same non-verified account. This procedure is analogous to the steps explained in Chapter 1, Section 1.3.1.

However, one drawback of this naïve method is that it often generates an extremely connected projection, especially with very rectangular graphs, i.e. graphs for which the dimension of the layer upon which we are projecting is much smaller compared to the dimension of the other layer, as in our case. For this reason, we employ the statistical procedure introduced in Chapter 1, Section 1.3.1 with the purpose of establishing the statistical significance of the number of common neighbours shared by a pair of nodes. The existence of each link is considered as a separate null hypothesis to test. Whenever two users share a number of common

neighbours that is significantly higher than the expected value of the null model, the hypothesis is rejected [38]. At the end of the validation procedure, only those pairs of users that share a *significantly* high number of interactions with the same set of uncertified nodes will be connected in the monopartite network of certified accounts.

Given the projected and validated network of retweets, we have performed a reshuffled community detection procedure, i.e. the Louvain algorithm [54] runs several times with a rearranged nodes' ordering and the partition with the highest modularity is selected; in this way we overcome the original algorithm's order dependence [55]. With this procedure we identify ten groups of non-isolated nodes. However, for the following analyses we focus on a subset of four of them, being those with a remarkable number of nodes (more than a hundred) and a non-trivial interpretation.

It is worth noting that, contrarily to the majority of studies on the same subject, the identification of clusters among verified users has not been studied simply attaching a suitable label to each unit. On the contrary, users in the same community share a significantly high number of non-verified users that have retweeted their content. Indeed, a validated edge between users i and j indicates that a high number of non-verified accounts who shared contents posted by i has also retweeted posts published by j and therefore they are considered similar by a majority of their audience and followers. Therefore, in our application, the behaviour of the non-certified accounts is the driver for the division in clusters.

3.3.2 Classification of non-verified users

Once a division in communities for the verified users has been detected, we try to extend the same division also to the group of non-verified users. In the following paragraphs we explain the procedures employed to perform this kind of analysis.

Polarization index

Once identified four groups of verified users, we want to analyse how the remaining accounts interact with them. The goal is to study whether the audience is polarized towards the source of information that better resembles their ideology or uniformly shares contents from accounts of different political orientation.

Indicate with \mathcal{C}_c for $c = 1, 2, 3, 4$ the sets of verified users identified at the end of the phase above. Also indicate with

$$\mathcal{V}_\alpha = \{i : i \in L \text{ and } m_{i,\alpha}^* = 1\} \quad (3.1)$$

the set of neighbours of the non-verified user $\alpha \in \Gamma$ in the bipartite network of verified/unverified users, i.e. the set of verified users node α has interacted with. The polarisation index for α is

$$\rho_\alpha = \max(\{\mathcal{I}_{\alpha,c} : c = 1, 2, 3, 4\}) \text{ for } \alpha \in \Gamma \quad (3.2)$$

with

$$\mathcal{I}_{\alpha,c} = \frac{|\mathcal{C}_c \cap \mathcal{V}_\alpha|}{|\mathcal{V}_\alpha|} \text{ for } c = 1, 2, 3, 4. \quad (3.3)$$

The term $\mathcal{I}_{\alpha,c}$ denotes the fraction of α 's interactions towards community c , i.e. the ratio of α 's neighbours belonging to community c . This index has the following characteristics: is bounded in $[0, 1]$, therefore $\rho_\alpha = 0$ means that no interaction has been observed with the four groups; values of ρ_α close to $1/4$ indicate that user α equally interacts with the four clusters; for all the other values, the greater ρ_α the higher the inequality in the number of interactions with the four communities (i.e. ρ_α close to 1 means that user α almost always has interacted with the same group).

Once a possible definition of polarization index has been provided, the goal is to employ it to extend the division in community also to the set of non-verified users. In other words, we “assign” to each non-verified user the label of the community with whom she/he interacted the most.

3.3.3 Influence analysis

In this Section we first propose a bipartite and directed representation of the tweet/retweet network. Then this representation will be employed to

project and validate the network onto the users layer, in order to identify pairs of users that significantly interact with the same tweets. Finally, we analyse the subgraphs made by the previously identified communities and isolate the most-retweeting and most-retweeted nodes.

Directed network of information flow

Consider the entire set of available users $U = L \cup \Gamma$ who tweeted and/or retweeted something during the overall time period. Then, consider the set of posts published on Twitter during T

$$P = \bigcup_{t \in T} P_t \text{ with } P_t = \{p : p \text{ is a tweet posted on Twitter during } t\}$$

and the sets of all tweets $\mathcal{T} \subseteq U \times P$ and retweets $\mathcal{R} \subseteq P \times U$ observed on the social network during T , i.e.

$$\mathcal{T} = \bigcup_{t \in T} \mathcal{T}_t \text{ and } \mathcal{R} = \bigcup_{t \in T} \mathcal{R}_t$$

where \mathcal{T}_t and \mathcal{R}_t are

$$\mathcal{T}_t = \{(i, p) : \text{user } i \text{ has tweeted post } p \text{ during } t\}$$

$$\mathcal{R}_t = \{(p, j) : \text{user } j \text{ has retweeted post } p \text{ during } t\}$$

respectively the sets of tweets and retweets observed in each single day. The number of nodes in the two sets U and P will be respectively indicated with the terms N_U and N_P . A directed edge $(i, p) \in \mathcal{T}$ indicates that user i is the original creator of post p , while an edge on the opposite direction $(p, j) \in \mathcal{R}$ shows that j has retweeted tweet p *at least one time* during the elections days T . Such a situation is shown in the left panel of Figure 18. In the picture, i is a user who publishes two posts p and q during the election days; user j retweets post p and user k retweets post q . Notice that, as already highlighted in Section 3.2, we cannot keep track of chains of sequential retweets. Therefore, the edge (q, k) does not necessarily mean that k has retweeted directly from i 's post: it is possible (less likely though) that k has retweeted post q from one of user i 's retweets. Another particular case is self-retweet: Twitter allows the users

to retweet their own posts, either directly from the original tweet or from somebody else's retweet. With this network representation, these cases can be illustrated as in right panel of Figure 18, where the retweet by i itself is indicated with the red arrow. However, this type of edges have been excluded from the analysis, since they represent a very small percentage of the overall number of links and we are rather interested in how people from the same political coalition interact with each other to boost the visibility of their opinion.

This bipartite and directed network $\mathbf{G}_{\text{BiD}}^*$ can be represented by means of two biadjacency matrices, one for the tweets $\mathbf{T}^* = \{t_{i,p}^* : i \in U \text{ and } p \in P\}$ and the other one for the retweets $\mathbf{R}^* = \{r_{p,j}^* : p \in P \text{ and } j \in U\}$. Essentially $t_{i,p}^* = 1$ if user i has tweeted post p and 0 otherwise and $r_{p,j}^* = 1$ whether or not at least one retweet to post p by user j is observed during T . As previously highlighted, the bipartite, directed network of tweeting and retweeting activity will be employed in the following Sections in order to detect the significant spreaders of Twitter viral contents.

Bipartite Directed Configuration Model

For the current analysis we are interested in determining which are the most pervasive tweet flows, discounting both the activity of users and the virality of tweets. Otherwise stated, we want to measure which are the non-trivial patterns observed in the network, not explained by the degree sequence for both layers, and describe the structure of the actual system. Therefore, let us represent the graph of tweeting and retweeting activity as a *bipartite and directed network* $\mathbf{G}_{\text{BiD}}^*$, in which the two layers collect the available users and tweets. Its non-symmetric adjacency matrix \mathbf{D} has the following structure

$$\mathbf{D} = \begin{pmatrix} \mathbf{0}_{N_U \times N_U} & \mathbf{T}_{N_U \times N_P}^* \\ \mathbf{R}_{N_P \times N_U}^* & \mathbf{0}_{N_P \times N_P} \end{pmatrix}, \quad (3.4)$$

where \mathbf{T}^* is the sub-matrix related to the tweets and \mathbf{R}^* the one for retweets. A user i tweeting post p is represented by the entry $t_{i,p}^* = 1$ of matrix \mathbf{T}^* . Analogously, the event of user j retweeting post p is represented by the entry $r_{p,j}^* = 1$ of the matrix \mathbf{R}^* .

The topology of this kind of graphs has been previously analyzed in [104] within the framework of a graph with core-periphery structure. However, a difference with the current case is that we have set the diagonal blocks equal to zero, because we cannot observe directed edges connecting solely users or tweets. In [104] the authors provide an entropy-base method to develop an unbiased null-model for this kind of graphs. The entire procedure is provided in Appendix B, here we just summarize the main steps. Let us use the following terms

$$k_i^{out}(\mathbf{T}^*) = \sum_{p \in P} t_{i,p}^* \quad \text{and} \quad k_i^{in}(\mathbf{R}^*) = \sum_{p \in P} r_{p,i}^*$$

to indicate respectively the out-going degree of users i , i.e. the number of her/his tweets that have been retweeted at least once, and her/his incoming degree, i.e. the number of posts that user i retweeted. Analogously, the number of retweets of post p is

$$k_p^{out}(\mathbf{R}^*) = \sum_{i \in U} r_{p,i}^*$$

while the number of authors of a generic post p is trivially

$$k_p^{in}(\mathbf{T}^*) = \sum_{i \in U} t_{i,p}^* = 1$$

for all $p \in P$, since each post cannot have more than one author.

Following the same steps of the previous section [105, 106] a null model for a bipartite directed network can be built simply considering the ensemble of bipartite directed graphs \mathcal{G}_{BiD} with the same number of nodes per layer as $\mathbf{G}_{\text{BiD}}^*$ and a probability distribution upon it. It can be shown that the probability distribution over the ensemble factorizes in two independent terms, both analogous to the BiCM, one related to the network of tweets and another one for the retweets, as follows

$$\begin{aligned} P(\mathbf{G}_{\text{BiD}} | \vec{z}, \vec{z}', \vec{z}') \\ = \left[\prod_{i,p} (q_{i,p})^{t_{i,p}} (1 - q_{i,p})^{1-t_{i,p}} \right] \cdot \left[\prod_{i',p'} (q'_{p',i'})^{r_{p',i'}} (1 - q'_{p',i'})^{1-r_{p',i'}} \right] \end{aligned} \quad (3.5)$$

where t_{ip} and $r_{p'i'}$ are the entries of the biadjacency matrices of the networks of tweets and retweets while

$$q_{i,p} = \frac{z_i \zeta_p}{1 + z_i \zeta_p} \quad \text{and} \quad q'_{p',i'} = \frac{z'_{i'} \zeta'_{p'}}{1 + z'_{i'} \zeta'_{p'}}$$

are the probability of existence for the entries $t_{i,p}$ and $r_{p',i'}$, respectively. Notice that $(\vec{z}, \vec{\zeta})$ and $(\vec{z}', \vec{\zeta}')$ are the N_U - and N_P -dimensioned Lagrange multipliers of the model and their numerical values can be determined from the data, choosing the values $(\vec{z}^*, \vec{\zeta}^*)$ and $((\vec{z}')^*, (\vec{\zeta}')^*)$ that solve the systems in equations (B.4) and (B.5) or equivalently that maximize the likelihood equations (B.3). However, because of the fact that each post has just one author, the Lagrangian multipliers of the tweets for \mathbf{T} are invariant for each node, i.e.

$$q_{i,p} = \frac{z_i^* \zeta^*}{1 + z_i^* \zeta^*}, \quad \forall p \in P.$$

Thus, imposing that the in-degree for all posts is always equal to 1, we get that

$$q_{i,p} = \frac{k_i^{\text{out}}(\mathbf{T}^*)}{N_P}, \quad \forall p \in P. \quad (3.6)$$

meaning that the probability that user i tweets post p simply depends on the number of other tweets posted by the same user. Let us remark that, due to the grand-canonical construction, this randomization also considers “non-physical” configuration, since the equality $\langle k_p^{\text{in}} \rangle = 1 \forall p \in P$ is enforced on average overall the ensemble and not pointwise for each configuration. The target of providing a more stringent null-model for tweeting will be the focus of future research. Moreover, in principle we could have considered the bipartite reciprocated configuration model presented in [104] but since we explicitly neglected the case of auto-retweet, there is no difference between the original model and the one employed here.

Validation of the projected directed network

The bipartite and directed network introduced in Section 3.3.3 is a simple, bipartite and directed representation of the tweet and retweet rela-

tions among the set of available users. Essentially, a directed edge (i, p) between user i and post p represents user i tweeting post p on the social network for the first time. By contrast, a directed edge (p, j) with $j \neq i$ denotes a retweet of user j to the same post p .

In this Section we aim at extending the projection and validation procedure proposed in the Introduction, Section 1.3.1, to the case of bipartite and directed networks. The extension is quite straightforward: the resulting monopartite network has the set of all users as nodes, whereas the set of directed potential edges is simply characterized by the set of pairs among all users. Then, the significance of each potential edge is evaluated.

Consider the potential directed edge (i, j) . As in the original mechanism, the quantity

$$V_{i,j}^* = \sum_{p \in P} t_{i,p}^* r_{p,j}^*$$

simply counts the number of i 's tweets that have been retweeted by j . In other words, $V_{i,j}^*$ identifies the number of times j has acted as a spreader for i , retweeting content posted by her/him. Note that, in general, $V_{i,j} \neq V_{j,i}$. Using the null model's expected value of this quantity we are able to infer the statistical significance of the tight between i and j . Since the probabilities per link are independent, the expected number of retweets from j to i 's posts is given by the following expression

$$\langle V_{i,j} \rangle = \sum_{p \in P} q_{i,p} q'_{p,j} \quad (3.7)$$

that is the parameter of the Poisson-Binomial random variable $V_{i,j}$ [39, 40]. Indeed, each $V_{i,j}$ is the sum of independent and Bernoulli-distributed variables, each of them with probability distribution

$$\begin{aligned} P(t_{i,p}^* r_{p,j}^* = 1) &= q_{i,p} q'_{p,j} \\ P(t_{i,p}^* r_{p,j}^* = 0) &= 1 - q_{i,p} q'_{p,j} \quad \forall p \in P. \end{aligned}$$

Notice that the previous term is the probability that j retweets some post p published by i . Due to the simplification of (3.6) the expected number

of V-motifs (3.7) can be rewritten as

$$\langle V_{i,j} \rangle = \frac{k_i^{out}(\mathbf{T}^*)}{N_P} \sum_{p \in P} q'_{p,j} = \frac{k_i^{out}(\mathbf{T}^*) k_j^{in}(\mathbf{R}^*)}{N_P}.$$

Therefore, the statistical significance of each directed edge (i, j) can be tested computing the following quantity

$$\text{p-value}(V_{i,j}^*) = \sum_{V_{i,j} \geq V_{i,j}^*} f_{PB}(V_{i,j})$$

for all $i, j \in P$, that simply implements the p-value of a Poisson-Binomial distribution [39, 40]. In other words, each directed link in the final network is treated as a null hypothesis to test and only those links associated to rejected hypotheses will be included in the final directed network. However, because of the large number of potential edges in our graph, we here employ a simple approximation of the quantity in the last equation, given by the following expression

$$\text{p-value}(V_{i,j}^*) = \sum_{V_{i,j} \geq V_{i,j}^*} f_{Poi}(V_{i,j}).$$

Essentially, we are simply approximating the p-value of a Poisson-Binomial distribution with the same quantity computed for a Poisson distribution, with the same parameter. The precision of such an approximation has been determined by Le Cam's Theorem, stating that

$$\sum_{V_{i,j}=0}^{N_P} |f_{PB}(V_{i,j}) - f_{Poi}(V_{i,j})| < 2 \sum_{p=1}^{N_P} (q_{i,p} q'_{p,j})^2.$$

In other words, the Poisson approximation is good whenever the expected number of successes is small.

The outcome of this procedure returns a squared $N_U \times N_U$ matrix collecting the significance of each link. At this point, each p-value is compared with a threshold value p_{th} obtained with the False Discovery Rate procedure proposed in [41] and revised in Subsection 1.3.1. Thus, the existence of each edge is treated as a separate null hypothesis to test

[38, 137] and only the links associated to rejected hypotheses have to be included in the validated network: whenever $\text{p-value}(V_{i,j}^*) \leq p_{th}$, j is considered a significant spreader of i 's tweets and the directed edge (i, j) is included in the validated network of information flows $\mathbf{F} = \{f_{i,j} : i, j \in U\}$.

In order to understand how viral news propagate through the validated network, we consider the division in communities detected in the first paragraph and analyse the percentage of retweets coming from inside of the same community and the percentage that, instead, derives from the other ones. In other words, given the division in clusters $\{\mathcal{C}_c : c = 1, 2, 3, 4\}$ identified with the previous steps, we define as $\mathcal{S}_i = \{j : f_{ij} = 1\}$ the set of spreaders for node i , i.e. the set of significant retweeters of node i . The significant information flow from node i towards each community is indicated as

$$\mathcal{F}_{i,c} = \frac{|\mathcal{C}_c \cap \mathcal{S}_i|}{|\mathcal{S}_i|} \text{ for } c = 1, 2, 3, 4, \quad (3.8)$$

that simply denotes the fraction of i 's retweets coming from community c . The distribution of these quantities for all nodes in the validated network may shed light on how the users distribute their attention patterns with respect to the tweets posted by the various alliances. The results of this kind of analyses will be discussed in the next Section.

3.4 Results

In this Section we discuss the results of the overall analyses performed on the dataset.

3.4.1 Groups of verified accounts

We here discuss the division in communities detected within the validated network of verified users, which construction has been described in Section 3.3.1.

Two blocks identify quite well the groups of Movimento 5 Stelle (from now on M5S) and right-leaning politicians. For instance, in the former

we find the accounts M5S Camera, M5S Senato, M5S Europa and Movimento 5 Stelle, as well as the politicians Danilo Toninelli or Luigi Di Maio. Instead in the latter we see the accounts Forza Italia, Lega - Salvini Premier, Gruppo FI Camera, Fratelli d'Italia, Noi con Salvini as well as the users Silvio Berlusconi, Matteo Salvini, Renato Brunetta or Giorgia Meloni. The two remaining communities are instead more heterogeneous. In one of them we find a high number of radios, newspapers or newscasts, such as Rai Radio 2, Radio 105, RTL 102.5, Tg Rai, Tg La7, Sky Tg 24, Rai News, la Repubblica, Il Corriere della Sera, Il Post. Since the majority of nodes refers to official accounts of news media, we characterize this group as the one collecting news spreaders and information channels. Finally, the last community encompasses some politicians within the left-leaning parties, such as Matteo Renzi and some other figures belonging to the Democratic party (included the account of the Partito Democratico itself). Therefore we use this interpretation for the last detected community.

Given the division in political alliances, we start analyzing the topological characteristics of the subgraphs made by each group. An important insight on users' behaviour comes from the observation of the hashtags used by verified users. Excluding the set of keywords used to extract the data, Figures 19 and 20 reproduce the hashtags used more frequently by the verified users of each community, selecting those with a frequency higher than or equal to the 0.5% of the total number of hashtags in the single community. All political (i.e. yellow, red and blue) communities have the name of their own party as the most mentioned tag; indeed we observe, respectively, the hashtags *m5s*, *pd* and *centrodestra* in their first position. Nevertheless, the second most used hashtag refers to the main opponent of the political alliance represented by the community. It is curious that this word is the Movimento 5 Stelle for both the right- and left-leaning alliances, since it was effectively the most voted party at the elections. Instead, the second most used hashtag by Movimento 5 Stelle is *renzi*, leader of the Partito Democratico at the moment of the elections, governing at that time. Also the major exponents of the other parties are mentioned, for instance *berlusconi*, *salvini* and *dimaio* appear in the left-

and right-leaning parties, as well as in the M5S one and the mixed group. Even if the frequencies of the hashtags are comparable, summarising, all alliances have the name of their own coalition as the most frequent one, followed by the names of the major competitors.

In addition to that, in the projected and validated network of verified users we also analyse the centrality of the involved users. In Appendix B we provide a list of the first fifty most central users in the validated network of retweets, together with their political affiliation with respect to the four identified groups. It is interesting to observe that the first positions are covered by medias and journalists. The majority of these figures are affiliated with the purple community, however we also observe users from the other political groups. For instance, we see *Il Fatto Quotidiano*, Peter Gomez and *IlSole24ORE* in the top ten most central users, the first two being affiliated with the M5S community, while the latter being classified into the left-leaning group.

3.4.2 Polarization of non-verified accounts

As stated in Section 3.3.2, the goal of introducing a polarization index was to extend the division in clusters obtained for the verified users also to the non-verified ones. In this Section we describe the results of such analysis.

The choice of the polarization index introduced in Section 3.3.2 has been mostly driven by the observation of users' interactions with the communities, graphically represented in top panel of Figure 21. Each square in the heatmap reproduces the average value of the quantity on the x -axis, computed over the set of non-verified users belonging to the community on the y -axis. Essentially, each square indicates the fraction of retweets directed towards each of the four listed communities (notice that the row-sum is not equal to one since a small amount of retweets has also been directed towards the remaining communities, not analysed in this work). Most of the non-verified users have an extremely unbalanced distribution of interactions with the members of the political alliances, registering the most numerous activity with other members of their own

community. Instead, in bottom panel of Figure 21 we represent the histogram of the polarization values obtained for the non-verified users. Many non-verified users show high values of polarization index, meaning that their attention patterns are mostly focused towards a limited group of political characters.

Figure 22 shows the biadjacency matrix of the bipartite network of verified and non-verified users. The coloured blocks identify the four communities in which the users have been classified: the red and blue blocks respectively identify the groups of left-leaning and right-leaning politicians; the yellow community collects the available political figures within the M5S party, while the violet group represents the information channels community. The rows of the matrix have been ordered according to the division in communities of the verified users, while the non-verified users have been sorted according to their political affiliation, i.e. the number of interactions towards each group, that is equivalent to the computation of the numerator of the term in equation (3.3) for all the available communities. Such a matrix exhibits a block structure along the diagonal, indicating a greater number of interactions towards the “preferred” community with respect to the others and therefore a higher density of links within the blocks with respect to the external density.

3.4.3 Influence analysis

At this point, we proceed with the identification of the significant sources of Twitter viral content. Inspired by the work of [78] we propose a bipartite and directed network of information flow, which construction has been described in Section 3.3.3. Very briefly, the users on the upper layer tweet and retweet the posts represented on the lower layer. As in [78], at first we simply project the original network of tweets and retweets onto the users layer: a directed edge between users i and j in the projected graph indicates that j has retweeted i ’s posts at least one time.

A list of the nodes with the highest in- and out-degrees per community can be found in Appendix B. The nodes with highest out-going degrees represent the users that have been retweeted most in the system,

while the nodes with the highest in-degree are those who retweet most. The symbol ✓ next to the name indicates that the user has been verified by the system. The top 20 most retweeted users in each group are mostly certified, because of the fact that, especially during the elections days, they are extremely active with the purpose of enlarging their pool of voters. Instead, the most retweeting users in each community are mostly uncertified. In this case, the symbols ✕ and • have been used to indicate respectively suspended users and users that cannot be found by the Twitter API. It is interesting that some of the most retweeting accounts have been suspended, by Twitter or by the other users: there is one suspended account in the M5S coalition and two others in the right-leaning one. Moreover, we also identify three accounts in the violet community that cannot be found by the Twitter API. However, as previously stated, this table only shows the first users with the highest in-degree. As a matter of fact, we have checked all the non-verified accounts with positive in- and out-degrees in the subgraphs generated by each community and we have recorded the percentage of suspended users: approximately 1% of the accounts have been suspended in the M5S community. This percentage slightly decreases in the information channels community and in the left-leaning one while increases in the right-leaning group, being the percentages respectively equal to 0.8% and 0.6% and 3%. The percentage of accounts that cannot be identified by the Twitter API are instead equal to 2% for the M5S and information channel communities while 3% for the left- and right-leaning communities.

However, these findings are not really explanatory in identifying the most effective users in the directed network of retweets, since we are missing a benchmark for stating if j is a significant contributor of the popularity of i 's posts. The identification of such significant ties will be performed following the steps described in Section 3.3.3.

A pictorial representation of the validated network of information flow is provided in Figure 23. Nodes' colour identifies the user's community while nodes' dimension indicates their out-degree in the validated graph. The structure of this network is better represented in Figures 24–27. Each plot focuses on the structure of the subgraphs of the directed

network generated by each community. Nodes dimension is directly proportional to their out-degrees in the subgraph, therefore the larger the node, the higher the number of times that user has been retweeted by the other accounts. Nodes colour is instead related to whether the account has been verified or not: blue for verified users, orange for non-verified ones.

The first plot is related to the M5S community and shows strongly connected block of (mostly non-verified) nodes that retweets among themselves and with the verified accounts of the community, the most central of which are the Twitter accounts of the newspaper *Il Fatto Quotidiano* and its journalists Marco Travaglio, Peter Gomez and Antonio Padellaro. In the other communities journalists and newspapers do not form such a strong core. It is interesting to notice that the M5S political leader Luigi Di Maio does not belong to this big community, but is located in a small community outside this large component.

The second plot represents the purple community. The most central nodes are the verified accounts of newspapers (see for example *La Repubblica* or *Il Corriere della Sera*) and information channels (such as *Sky TG24*, *Tg La7*, *Agenzia Ansa* or *Rainews*). Some politicians such as Pietro Grasso or Giuseppe Civati are present in this group, together with the political parties of *Rifondazione* and *Potere al Popolo*. These politicians represent the most extreme, left-leaning orientation and that have not encountered a commonality of interests and supporters with the accounts in the community of *Partito Democratico* and indeed they belong to different communities.

In the plot associated to the left-leaning community (the third one) we identify a central block of mostly non-verified users. The most retweeted figures are Matteo Renzi and the account of *Partito Democratico*, as shown by their high values of out-degrees. The remaining verified nodes are mostly well-known characters in the political scenario (such as Maria Elena Boschi and Carlo Calenda), as well as newspapers (see for example *Il Foglio* or *IlSole24Ore*). Among the non-verified users we have the accounts of the *Partito Democratico* political parties related to the areas of Milan and Rome.

Finally, the plot associated to the right-leaning community is characterised by two quite separate clusters; one of them is centered on the accounts of people belonging to Lega Nord, such as Matteo Salvini, Claudio Borghi and the party Lega-Salvini Premier. On the other side there are the accounts of Forza Italia and Gruppo FI Camera and some of its exponents like Silvio Berlusconi or Renato Brunetta. The two verified nodes of Giorgia Meloni and Fratelli d'Italia (the political party she is leading) receive retweets from both sides, nevertheless being closer to the Lega pole. Another popular node is CasaPound, that has its own circle of retweeters and share some interactions with the subgroup of Lega Nord.

In order to understand how viral news propagate through the validated network, we analyse the percentage of retweets coming from inside of the same community and the percentage that, instead, derives from the other ones. The results are shown in Figure 28. Despite the fact that the distribution appears fuzzier after the validation, we still notice that a higher number of retweets comes from users from the same community. Even after the validation, communities three and four appear well connected and there are less interactions from other groups. Instead, we observe a higher number of interactions among communities one, two and three. The interactions between two and three confirm what stated before: in addition to the information channels, the former collects the political figures with left-oriented ideology but that have not encountered a commonality of interests and supporters with the accounts in the first community. Moreover, the interactions with group two are due to the fact that this group collects the news spreaders and information channels and it is plausible that accounts from different communities have shared their contents.

3.5 Discussion and conclusions

This work is focused on the identification of the sources of Twitter viral content during the last Italian electoral campaign of 2018. We have gathered the data from the Twitter API during the month before the elec-

tions. As a first step, we have exploited the way people consume news on the social media to identify four groups of strongly connected verified users: two certified users are connected in the validated network of retweets if a significantly high number of non-certified people retweets content published by both of them. By construction, the behaviour of non-certified accounts is thus exploited to understand the division of the political sphere into clusters.

An analysis of the obtained communities shows that the most central accounts are mostly newspapers, journalists and information channels in general, each of them belonging to one of the previously listed clusters. Looking at the hashtags included in the posts published by the groups, we manage to interpret these coalitions and we also see that the most used keywords in each alliance are referred to the party itself and its members, followed by keywords related to political competitors.

Given the obtained division in political alliances, we have studied the behaviour of the remaining non-verified users towards these groups. More specifically, we have observed the fraction of retweets directed towards each coalition: we observe a strongly polarized behaviour, since the majority of the uncertified accounts in the bipartite network of retweets mostly interacts with one community only. In order to strengthen this result, we also perform a different analysis comparing the distribution of polarization values observed for users with the same number of interactions. Also in this case, the distribution is skewed towards the higher values, indicating a focus towards the same group of users. Refer to Appendix B for a pictorial representation of this phenomenon.

As a second step, we have focused our attention on the identification of significant news spreaders. Following the method presented in [78] we have constructed the directed network of retweets among users and we have selected the names with the highest out-degree and in-degree. Many of the former are verified, since they are extremely active during the electoral campaign in order to enlarge their pool of voters. Instead, the latter are mostly non-certified users, some of which have also been suspended by Twitter or by some other users.

In order to validate our findings, we have constructed the bipartite

and directed network of tweets and retweets introduced in Section 3.3.3 and performed the validation procedure explained in the same Section. The outcome of this analysis is a monopartite, directed network of users, in which an edge from i to j indicates that the latter has retweeted contents posted by the former a significantly high number of times (i.e. where the significance is evaluated through the null model that takes into account the information regarding the number of tweets and retweets posted by each user and the number of retweets received by any post). The visualization of this validated network helps in understanding the actual composition of each coalition, as well as the possible interconnections between them. For instance, we have observed that the majority of the connections between one community and another one happens between verified and unverified users (where probably the latter has retweeted some posts coming from the former). However, we also see connections involving newspapers and information channels belonging to different coalitions, confirming again their essential role and centrality in spreading news on the social networks. Finally, we have analyzed the origin of the retweets received by each community: even though the distribution seems less polarized in this case, we clearly see that a higher percentage of the retweets received by a community comes from users belonging to the community itself, showing that most of the interactions still derive from the same sphere of influence.

In our view, the methodological contributions of this work are manifold. A first contribution resides in the fact that, at odds with a majority of other works dealing with the same topic, our dataset was not manually labelled: we identified groups of strongly connected users starting from the behaviour of non-certified ones and how they interact with the certified users. Despite this data-driven approach for classifying the accounts, we managed to identify four clusters of users that are closely aligned with the Italian political division. Therefore, how people consume the news and interact with the main political figures helps to shed light on the actual division of the verified users according to their political orientation.

Our second contribution resides in the representation of the network

of activities on Twitter as a bipartite, directed network. We employed the null model proposed in [104] to identify the significant information flows between pairs of users and different communities. Using the same method, it would be interesting to analyze the users that significantly retweets their own contents, to boost their visibility. Other interesting lines of research, left for future investigation, include a validation of our approach to a corpus of different elections, in order to identify potential regularities or differences between countries, and the use of tools from natural language processing to infer how positively- and negatively- connoted tweets are distributed across the communities.

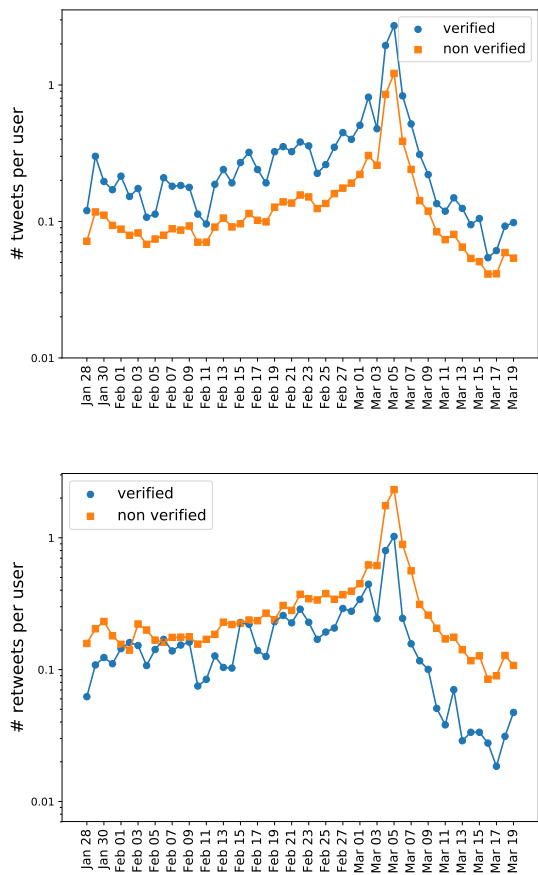


Figure 17: Number of tweets and retweets per user per day, considering separately the number of tweets posted by verified and non-verified accounts.

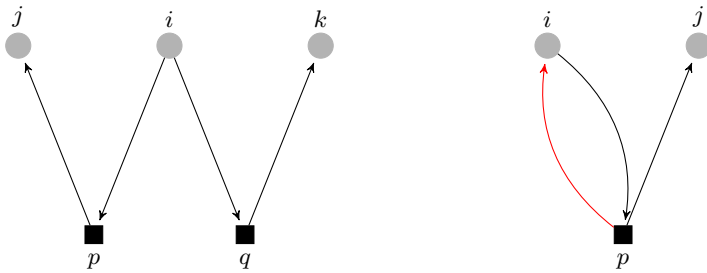


Figure 18: Left: User i is a central user and publishes two posts p and q during the elections. Users j and k respectively retweet the same posts p and q at least one time. Right: illustration of the case in which i retweets one of her/his own posts (the red arrow). This case is excluded from the analysis.

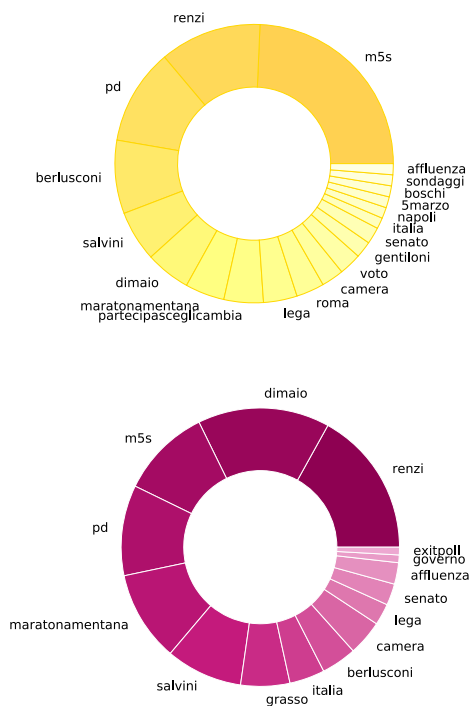


Figure 19: Hashtags with frequency higher than 0.5% of total number of hashtags used by the groups of verified users in M5S and information channels communities, excluding “elezioni”, “elezioni2018”, “elezionipolitiche”, “4marzo”, “4marzo2018”, “marzo2018”, “politiche”, “politiche2018”, “elezionipolitiche2018”, “elezioni4marzo2018”. The represented color scales have been obtained rescaling the original frequencies between 0 and 1.

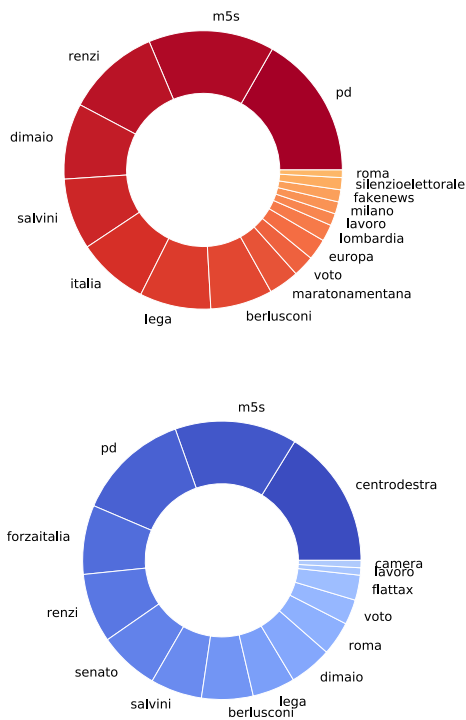


Figure 20: Hashtags with frequency higher than 0.5% of total number of hashtags used by the groups of verified users in left- and right-leaning communities, excluding “elezioni”, “elezioni2018”, “elezioniipolitiche”, “4marzo”, “4marzo2018”, “marzo2018”, “politiche”, “politiche2018”, “elezionipolitiche2018”, “elezioni4marzo2018”. The represented color scales have been obtained rescaling the original frequencies between 0 and 1.

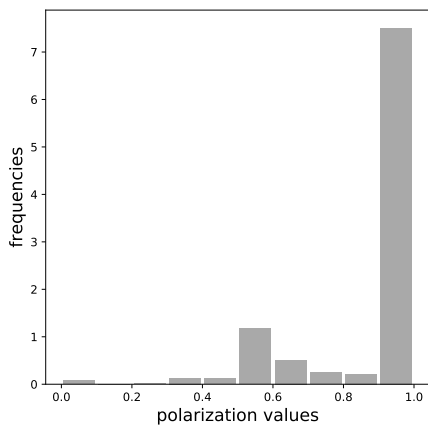
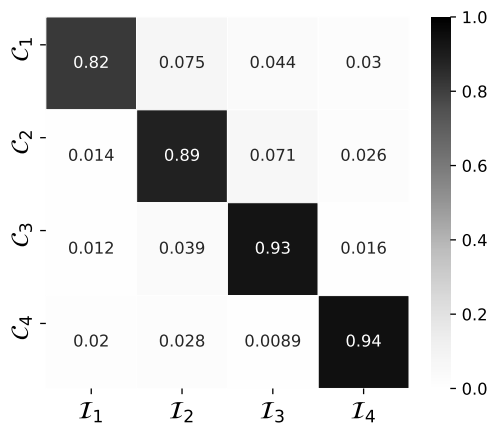


Figure 21: Top: heatmap reproducing the average fraction of interactions towards each community, for subsets of non-verified users. Each square in the heatmap indicates the average value of the quantity reported on the x -axis, computed over the set of non-verified users belonging to the community on the y -axis. Bottom: histogram of obtained polarization values.

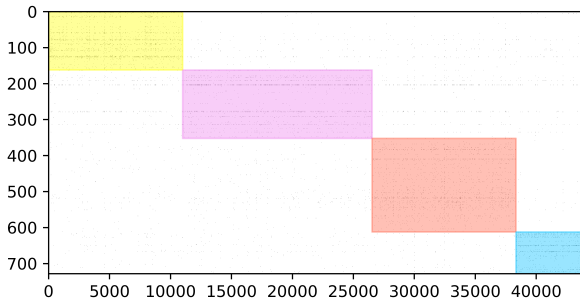


Figure 22: Biadjacency matrix of the network of retweets between verified and non-verified users. The coloured blocks identify the four communities of verified users obtained with the community detection method: yellow for M5S, red and blue respectively for left- and right-leaning alliances and purple for the information channels community. The rows of the matrix have been ordered according to the division in communities, while the columns are sorted according to the political affiliation of non-verified users, i.e. the number of interactions with each group.

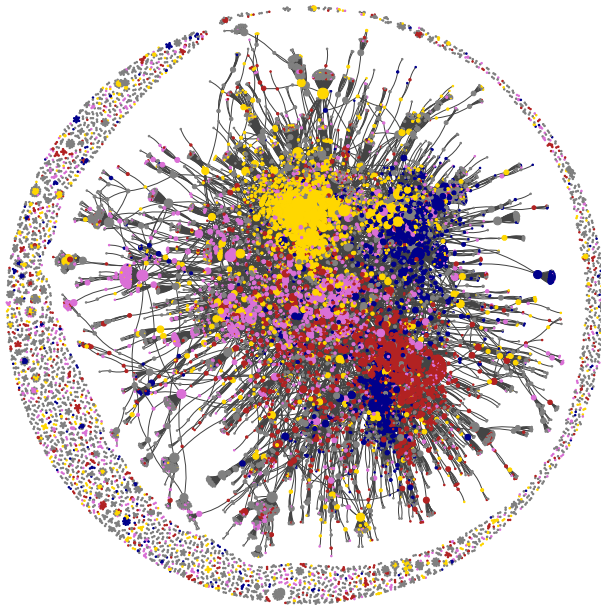


Figure 23: Validated and directed network of retweets. Entire directed network of retweets after the bipartite validation procedure. An edge from i to j indicates that j has retweeted i a significantly high number of times. Nodes' colour identifies the community to which the user belongs (red for left-leaning, yellow for M5S, purple for information channels and blue for right-leaning communities) while nodes' dimension indicates their out-degree in the validated graph (i.e. how often their posts have been retweeted a significantly high number of times). Note that the blue nodes are divided into 2 subcommunities, one closer to the M5S community, the other closer to the left-leaning one.

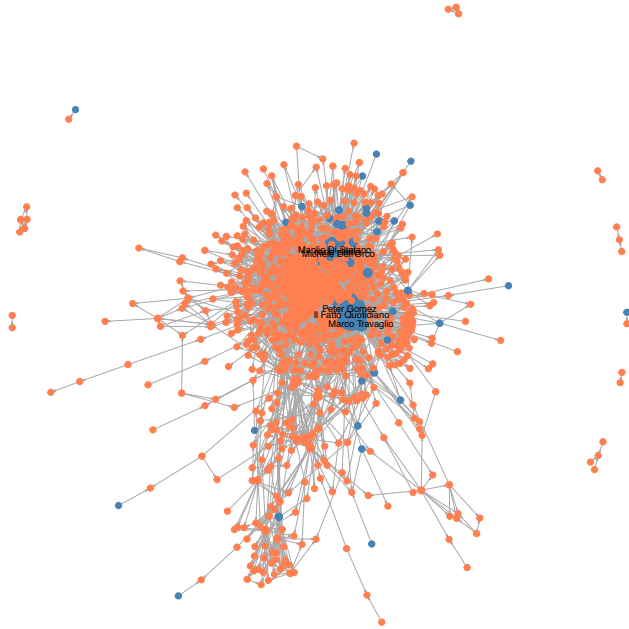


Figure 24: Most retweeted users in group \mathcal{C}_1 . Nodes' dimension is proportional to the out-degree in the subgraph generated by the community (i.e. number of retweets received from users from the same community). The blue color indicates that the node is verified, while orange is not verified.

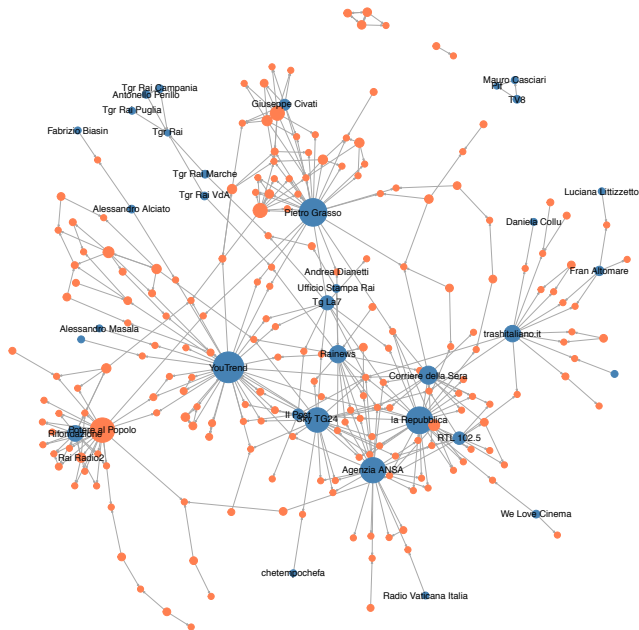


Figure 25: Most retweeted users in groups C_2 . Nodes' dimension is proportional to the out-degree in the subgraph generated by the community (i.e. number of retweets received from users from the same community). The blue color indicates that the node is verified, while orange is not verified.

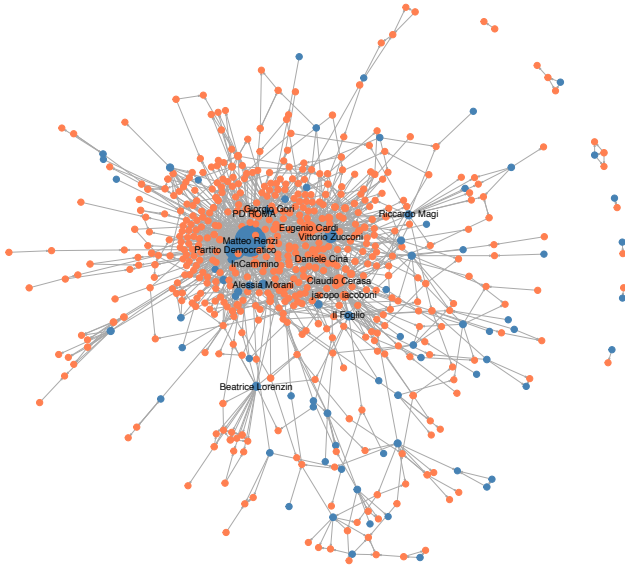


Figure 26: Most retweeted users in groups \mathcal{C}_3 . Nodes' dimension is proportional to the out-degree in the subgraph generated by the community (i.e. number of retweets received from users from the same community). The blue color indicates that the node is verified, while orange is not verified.

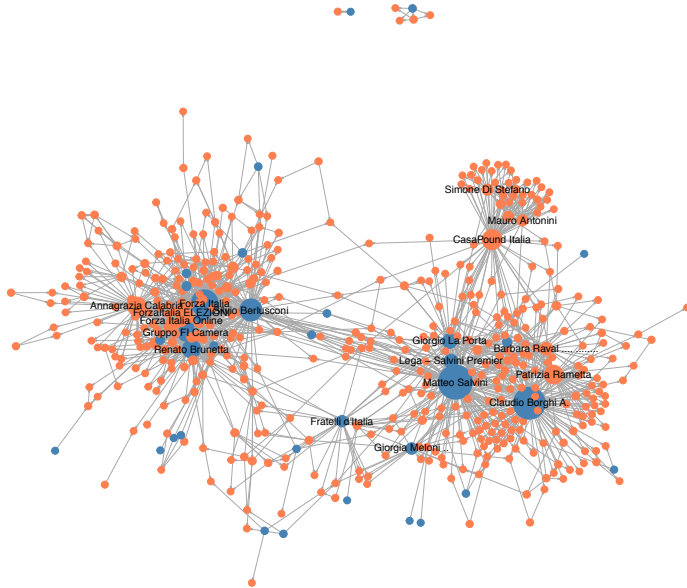


Figure 27: Most retweeted users in groups C_4 . Nodes' dimension is proportional to the out-degree in the subgraph generated by the community (i.e. number of retweets received from users from the same community). The blue color indicates that the node is verified, while orange is not verified.

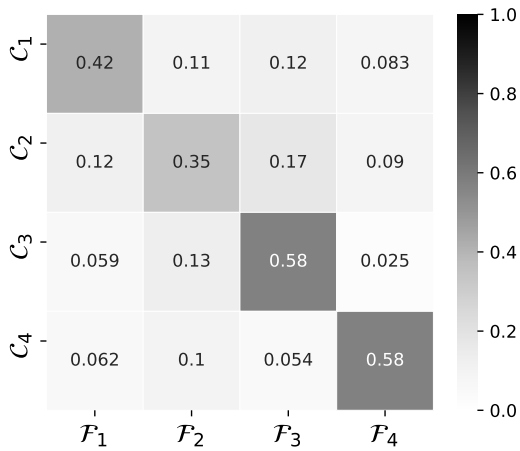


Figure 28: Heatmap reproducing the average number of retweets coming from users belonging to each community. Each square undicates the average value of the quantity reported on the x -axis, computed over the set of users belonging to the community on the y -axis.

Chapter 4

Collaboration and followership: a stochastic model for activities in social networks

This Chapter is a joint work with my supervisors Prof. Irene Crimaldi and Dr. Fabio Saracco. The full text of the article is also available from the arXiv repository, preprint number 1811.00418.

Abstract

In this work we investigate how future actions are influenced by the previous ones, in the specific contexts of scientific collaborations and friendships in online social networks. We are not interested in modeling the process of link formation between the agents themselves, we instead describe the activity of the agents, providing a model for the *formation of the bipartite network of actions and their features*. Therefore we only require to know the chronological order in which the actions are performed and not the order in which the agents are observed. Moreover, the total number of possible features is not specified a priori but is allowed to increase along time, and new actions can independently show some new-entry features or exhibit some of the old ones. The choice of the old features is driven by a degree-fitness method. With this term we mean that the probability that a new action shows one of the old features does not solely depend on the “popularity” of that feature (i.e. the number of previous actions showing it) but is also affected by some individual traits of the agents or the features themselves, synthesized in certain quantities called “fitnesses” or “weights”, that have different shapes and meaning according to the specific setting considered. We show some theoretical properties of the model and provide statistical tools for the parameters’ estimation. The model has been tested on three different datasets and the numerical results are provided and here discussed.

4.1 Introduction

This Chapter is introduced into the stream of literature regarding stochastic models for the formation and evolution of bipartite networks. This topic finds its root into the well-known preferential attachment model proposed in [110], in which the authors provide an explanation of the power-law distribution of node degrees in the World Wide Web (WWW). The success of their proposal resides in the fact that only the simple preferential attachment rule that drives the formation of links is able to reproduce, with good accuracy, the degree distribution of many real-world networks. However, successive works enriched the model with other ingredients in order to overcome the “first move advantage” (i.e. the fact that older nodes have greater degrees by construction), thus permitting to younger nodes to easily grow. For instance, the authors in [29, 30] introduce the fitness, a quantity defined at node level that measures the intrinsic ability of the vertex to collect links. This new variable has been introduced with a multiplicative effect on nodes’ degrees, therefore its role is to amplify or dampen the preferential attachment contribution. Later [31] has extended the potential of the fitnesses, building a generative model that solely embeds fitnesses and not node degrees. In this case, a fitness variable is assigned to every node, representing an intrinsic property of the vertex in the system. Then, a link between a pair of nodes is expressed as a function of the nodes’ fitnesses only. By modifying their distribution, fitnesses only are able to reproduce the power-law degree distributions present in many networks.

The main novelty of the previous works was not in the specification of the model per se, but in providing an explanation for the structure of the examined networks. Indeed, in most of the fitness methods, some attributes of the nodes not directly observed in the network define the structure of the graph itself. For instance, [32] introduces fitnesses that vary over time, giving an explanation to the limited in time growth in citation of most of the papers.

All previous efforts were devoted to monopartite, directed or undirected, networks. A much smaller number of contributions is available

for the description of the evolution of bipartite networks. Some examples in this direction have been proposed for specific datasets. For instance, in [121] the authors propose a generative model to study the bipartite network of lawyers and clients that develops according to a recommendation process: more popular attorneys are also more likely to be hired by new clients. Furthermore, the authors in [122] provide a framework in which the simultaneous evolution of two systems has been studied. Indeed, they analyse communities of scientists considering both the monopartite network describing the interactions among agents and the bipartite semantic network in which the agents are associated to the concepts they use. Another example is [123], in which the structure of the (growing) bipartite trade network was reproduced by assigning links with sequential preferential attachment. In order to describe the generation of an innovative product, following the idea of the “adjacent possibles” [124], new nodes (i.e. new products) are derived by the structure of an unobserved monopartite network of products describing the hierarchical productive process relations. Therefore, the evolution of the bipartite system is due to the simultaneous dynamics of an unobserved evolving network.

Following the stream of literature regarding network models based on a latent attribute structure, a new model was introduced in [125]. In this context, nodes sequentially join the considered network, each of them showing a set of features. Each node can either exhibit new features or adopt some of the features already present in the network. This choice is regulated by a preferential attachment rule: the larger the number of nodes showing a certain feature, the greater the probability that future nodes will have it too. The total number of possible features is not specified a priori but is allowed to increase in time. Differently from [119, 126], each node has been assigned a fitness variable that accounts for nodes’ personal ability to transmit its own features to future nodes. Starting from here, the model in [127] introduces some novelties in the previous context: the probability to exhibit one of the features already present in the network is defined as a mixture (i.e. a convex combination) of random choice and preferential attachment. However, neither fitnesses nor

weights are introduced in the model, so that all nodes are assumed to have equal capabilities in transmitting their personal features to the newcomers.

The present work moves along the same research line of the previously mentioned papers [125, 127] but with a different spirit. The main drawback of these two models resides in the assumed chronological order of nodes' arrivals, which may typically be unknown (or non-relevant) in many real-world systems. In the present paper we overcome this limitation: given a system of n agents, we provide a *model for the formation of the bipartite network of agents' actions and their features*. Therefore, this model can also be applied to all settings in which the agents of interest are not observed in a specific chronological order, because the assumption on the chronological order is specified on the agents' actions only. Furthermore the probability to exhibit one of the features already observed is defined as a mixture of random choice and "*preferential attachment with weights*", i.e. the probability of connection depends both on the features' degrees and the fitness of the agents involved and/or of the features themselves. These weights can have different shapes and meaning according to the specific setting considered: the weight at time-step t of the observed feature c can depend on some characteristics of c itself, or it can be directly established by the agent performing action t , or it may represent the "inclination" of the agent performing action t in adopting the previous observed features, or it may implicitly due to some properties of the agent performing the previous action j with c among its features (for instance, her/his ability to transmit her/his own features). We analyse two datasets of scientific publications (respectively arXiv for Theoretical High Energy Physics, or more briefly Hep-Th, and IEEE for Automatic Driving) and a dataset of posts of Instagram. We not only obtain a good fit of our model to the data, but our analysis also results useful in order to highlight interesting aspects of the activity of the three considered social networks.

The rest of the Chapter is so organized. In Section 4.2 we illustrate in detail the proposed model for the formation of the actions-features bipartite network. In Section 4.3 we explain the meaning of the model param-

eters and the role of the weights introduced into the preferential attachment term. In Sections 4.4 we describe the general methodology used to analyse the data, while in Section 4.5 we describe the three datasets employed to test the model. We describe the results of the application of the model to the data in Section 4.6 while we summarize the overall contents of the work and recap the main findings in the last Section 4.7. The material collected in Appendix C is the following: some asymptotic results regarding the behavior of the total number of features and the mean number of edges in the actions-features bipartite network are collected in Subsections C.1-C.2, while Subsection C.3 contains a description of the statistical tools used for the estimation of the model parameters.

4.2 The model

Suppose to have a system of n agents that sequentially perform actions along time. Each agent can perform more than one action. In this framework, the running of time coincides with the flow of the actions and so sometimes we use the expression “time-step t ” in order to indicate the time of action t . Each action is characterized by a finite number of features and different actions can share one or more features. Notice that we do not specify a priori the total number of possible features in the system but we allow this number to increase in time. In what follows, we describe the model for the dynamical evolution of the bipartite network that collects actors’ actions on one side and the corresponding features of interest on the other side. We denote by \mathbf{M} the biadjacency matrix related to this network.

The dynamics starts with the observation of the first action, performed by an agent of the considered system, that shows N_1 features, where N_1 is assumed Poisson distributed with parameter $\alpha > 0$. This distribution will be denoted from now on by the symbol $\text{Poi}(\alpha)$. We number the observed features with the index c from 1 to N_1 and we set $m_{1,c} = 1$ for $c = 1, \dots, N_1$.

It is worth noting that, in order to adapt the notation to the one introduced in Chapter 1, the term N_t should be used to denote the total num-

ber of nodes present in the system at time t . However, the running of the time-steps of the model coincides with the flow of the actions, thus the number of actions performed at each time-step is a deterministic quantity always equal to 1. Therefore, I decided to employ a slight abuse of notation and use the term N_t to identify the number of *new features* introduced in the network at time t (i.e. the number of new nodes in one of the layers only), since the total number of nodes observed at time t on the two layers of the actions-features network can be easily recovered as a function of this quantity, as follows

$$t + \sum_{j=1}^t N_j \text{ for } t \geq 1. \quad (4.1)$$

Moreover, I will denote with L_t the quantity

$$L_t = \sum_{j=1}^t N_j \quad (4.2)$$

that is the total number of different features observed until time t . As a matter of fact, in Chapter 1 of this thesis we used L to denote the set of nodes on one layer only in a bipartite network and N_L to indicate its cardinality. However, in this case I will use the symbol L for the second quantity, in order to keep the notation as simple as possible due to the time-dependence that we are about to introduce.

Then, for each consecutive action $t \geq 2$, we have:

1. Action t exhibits some old features, meaning that these features have already been shown by some of the previous actions $\{1, \dots, t-1\}$. More precisely, the new action t can independently display each old feature $c \in \{1, \dots, L_{t-1}\}$ with probability

$$p_{t,c} = \frac{\delta}{2} + (1 - \delta) \frac{\sum_{j=1}^{t-1} m_{j,c} w_{t,j,c}}{b_t} \quad (4.3)$$

where $\delta \in [0, 1]$ is a parameter of the model, $m_{j,c} = 1$ if action j shows feature c and $m_{j,c} = 0$ otherwise and $w_{t,j,c} \geq 0$ is the random weight associated to feature c measured at the time of action t , that

can be related to the course of previous actions j . Finally b_t is a suitable normalizing factor ensuring that the term $\sum_{j=1}^{t-1} m_{j,c} w_{t,j,c} / b_t$ belongs to $[0, 1]$. We will refer to quantity (4.3) as the *inclusion probability* of feature c at time-step t .

2. Action t can also exhibit a number of new features N_t , where N_t is assumed $\text{Poi}(\lambda_t)$ -distributed with parameter

$$\lambda_t = \frac{\alpha}{t^{1-\beta}}, \quad (4.4)$$

where $\beta \in [0, 1]$ is a model parameter. The variable N_t is supposed independent of $\{N_1, \dots, N_{t-1}\}$ and of all the appeared old features and their weights (including those of action t).

With the observation of action t , all the matrix elements $m_{t,c}$ with $c \in \{1, \dots, L_t\}$ are set equal to 1 if action t shows feature c and equal to 0 otherwise.

Example. Here is an example of a \mathbf{M} matrix with $t = 3$ actions:

$$\mathbf{M} = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}.$$

In grey we highlight the new features for each action: we have that the first three actions show respectively $N_1 = 4$, $N_2 = 2$, $N_3 = 3$ new features, and so the total number of characteristics at each timestep becomes $L_1 = 4$, $L_2 = 6$ and $L_3 = 9$. For each action t , we have $m_{t,c} = 1$ for each $c \in \{L_{t-1} + 1, \dots, L_t\}$. Moreover, some elements $m_{t,c}$, with $c \in \{1, \dots, L_{t-1}\}$, are equal to 1 and they represent the features brought by previous actions exhibited also by action t .

It may be worth noting that our model resembles the one known as the Indian Buffet Process in Bayesian Statistics [128, 129, 130], but indeed there are significant differences in the definition of the inclusion probabilities: in particular, the main differences reside in the mixture parameter δ and the weights $w_{t,j,c}$. Moreover, Bayesian Statistics deals with exchangeable sequences, while here we do not require this property. As a consequence, the role played by each parameter in (4.3) and (4.4) results more straightforward and easy to be implemented.

4.3 Discussion on the model

We now discuss the meaning of the model parameters α , β and δ and the role of the random weights $w_{t,j,c}$. Appendix C provides asymptotic results regarding the total number of features and edges observed in the network, as well as the statistical tools employed to estimate the model parameters.

4.3.1 The model parameters

In the above model dynamics, the probability distribution of the random number N_t of new features brought by action t is regulated by the pair of parameters (α, β) , as stated in equation (4.4). Specifically, the larger α the higher the total number of new features brought by an action, while β controls the asymptotic behavior of the random variable L_t in equation 4.2, i.e. the total number of features observed for the first t actions, as a function of t .

Analysing equation (4.3) of the above model dynamics, we see that, for a generic action t , both the parameter δ and the random weights $w_{t,j,c}$ affect the number of old features ($c = 1, \dots, L_{t-1}$) also shown by action t . Specifically, the value $\delta = 1$ corresponds to the *pure i.i.d. case* with inclusion probability equal to $1/2$: an action can exhibit each feature with probability $1/2$ independently of the other actions and features. Instead the value $\delta = 0$ corresponds to the case in which the inclusion probability $p_{t,c}$ entirely depends on the (normalized) total weight associated to feature c at the time of action t , i.e. to the quantity

$$\frac{\sum_{j=1}^{t-1} m_{j,c} w_{t,j,c}}{b_t}. \quad (4.5)$$

In equation (4.5), the term $w_{t,j,c} \geq 0$ is the random weight at time-step t associated to feature c that can be related to the course of previous actions j . We denote this case as the *pure weighted preferential attachment case* since the larger the total weight of feature c , the greater the probability that also the new action will show feature c . Finally, when $\delta \in (0, 1)$, we have a mixture of the two cases above: the smaller δ , the more significant is

the role played by the weighted preferential attachment in spreading the observed features to new actions. In the sequel we will refer to (4.5) as the *weighted preferential attachment term*.

4.3.2 Meaning of the random weights

Regarding the weights, the possible ways in which they can be defined benefit of a great flexibility. Of course their meaning has to be discussed in relation to the particular application considered. For instance, the weight $w_{t,j,c}$ can be directly assigned by the agent performing action t to feature c in connection with the previous action j , or it may represent the inclination of the agent performing action t of adopting the previous observed features, or it may implicitly due to some properties of the agent performing the previous action j (for instance, her/his ability to transmit her/his own features). We here describe some general interesting frameworks:

- a. **Constant weights.** By setting $w_{t,j,c} = 1$ for all t, j, c with normalizing factor $b_t = t$, then all the observed features have the same weight. Then the sum in the numerator of (4.5) becomes the *popularity* of feature c , the total number of previous actions that have already exhibited feature c . Therefore, the quantity in equation (4.5) essentially represents its *average popularity* (we divide by t instead of $t - 1$ in order to avoid the quantity (4.5) to be exactly equal to 1 for all the first N_1 features). In this case the actions-features dynamics coincides with the nodes-features dynamics considered in [127].
- b. **Weights depending on past actions.** Assume that a positive random variable x_i (with $i = 1, \dots, n$) is associated to each agent in order to describe her/his ability to transmit the features of her/his actions to the others. This random variable can be interpreted as a static *fitness* as defined in [29, 30, 31]. In this case the weights $w_{t,j,c}$ can be defined as $x_{i(j)}$ or a function of this quantity, where $i(j)$ denotes the agent performing action j . In particular, we have

$w_{t,j,c} = w_j$ so the weights only depend on j . Hence, the weight of a feature c is only due to the fitness of the agent that performs an action with c among its features and the sum in the numerator of (4.5) becomes the total weight of feature c due to the agents that have previously exhibited it in their actions. The quantity

$$b_t = \kappa + \sum_{h=1}^{t-1} w_h$$

can be chosen as normalizing factor, i.e. we basically normalize by the total fitness of the agents that have performed actions $1, \dots, t-1$. Notice that the previous case can be seen as a special case of the present one, taking $x_i = 1$ and $\kappa = 1$. Moreover, another interesting element to observe is that the weighted preferential attachment term (4.5) can be explained with an urn process, as follows.

Example. For each feature c , let $t(c)$ be the first action that has c as one of its features and imagine to have an urn with balls of two colors, say red and black, and associate an extraction from the urn to each action $t \geq t(c) + 1$. The initial total number of balls in the urn is $\kappa + \sum_{h=1}^{t(c)} w_h$, of which $w_{t(c)}$ red. At each time-step $t \geq t(c) + 1$, if the extracted ball is red then action t exhibits feature c and the composition of the urn is updated with w_t red balls; otherwise, action t does not exhibit feature c and the composition of the urn is updated with w_t black balls. Therefore quantity (4.5) gives the probability of extracting a red ball at time-step t . This is essentially the nodes-features dynamics considered in [127] with $\delta = 0$ only.

If we have $x_i \leq 1$, an alternative normalizing factor is $b_t = t$. In this case the quantity (4.5) is the empirical mean of the random variables $m_{j,c} w_j$, with $j = 1, \dots, t-1$ (again we divide by t instead of $t-1$ for the same reason explained above).

- c. Weights depending on past actions and on time.** Case b. can be extended to consider fitness variables that change along time, so

that we have $w_{t,j,c} = w_{j,t}$ defined in terms of $x_{i(j),t}$ where $i(j)$ denotes the agent that performs action j and $x_{i(j),t}$ is her/his fitness at the time-step of action t , thus following prescription similar to those of [32, 118]. We can also extend it to the case in which the actions can be performed in collaboration by more than one agent. In this case the weight $w_{j,t}$ can be defined as a function of the fitness at time-step t of all the agents performing action j .

- d. Weights depending on features and on time.** We can set $w_{t,j,c} = w_{t,c}$ for all j, c, t with $b_t = t$ so that the term (4.5) becomes the average popularity of feature c adjusted by the quantity $w_{t,c}$. For instance, we can choose $w_{c,t}$ as a decreasing function of $t^*(c) = \max\{j : 1 \leq j \leq t-1 \text{ and } m_{j,c} = 1\}$ that is the last action before action t that has c among its features. By doing so, the average popularity of c in equation (4.5) is discounted by the length of time between the last appearance of feature c and t . Another possibility could be to choose a weight $w_{t,c}$ in order to give more relevance to the features already shown by the same agent performing action t in the previous actions. More precisely, we can denote by $i(j)$ the agent that performs action j and, for each action t , we can define $w_{t,c}$ as an increasing function of the sum $\sum_{j=1, \dots, t-1, i(j)=i(t)} m_{j,c}$ so that the more an agent has exhibited feature c in her/his own previous actions, the greater the probability that also her/his new action will show feature c . An additional possibility is to eliminate the dependence on t and consider weights $w_{t,j,c} = w_c$, where w_c can be seen as a “fitness” random variable associated to feature c .
- e. Weights depending on time.** We can modify case b. giving a different meaning to x_i . Indeed, we can associate to each agent i a positive random variable x_i in order to describe her/his inclination of adopting the already appeared features. Then we can define the weight $w_{t,j,c}$ as $x_{i(t)}$ or a function of it, where $i(t)$ denotes the agent performing action t . In this way, we have $w_{t,j,c} = w_t$ for all j, c, t , that is the weights only depend on the “inclination” of the agent performing the action and, if we set $b_t = t$ as in case d., the term

(4.5) becomes the average popularity of feature c adjusted by the quantity w_t .

- f. Weights depending on features and past actions.** Finally, we can take $w_{t,j,c} = w_{j,c}$ (i.e. depending on j and c , but not on t) in order to represent the weight given by the agent performing action j to feature c exhibited in this action. Therefore the total weight of feature c at time-step t is the total weight given to feature c by the agents who performed the previous actions.

These are just general examples of possible weights. We refer to the following applications to real datasets for special cases of the above examples. It is worth noting that the weights $w_{t,j,c}$ may be not independent. For example, given case e. we have exactly the same weight for all the actions performed by the same agent.

4.4 General methodology for the analysis

We here provide a detailed outline of the performed analyses used for the considered case studies.

4.4.1 Estimation of the model parameters

For each application we sort the considered activities in chronological order, the features are instead sorted according to their arrival time in the system. Then we construct the features matrix \mathbf{M} , which entries are equal to $m_{t,c} = 1$ if action t shows feature c and $m_{t,c} = 0$ otherwise. We provide the estimated values of the parameters α, β, δ of the model by means of the tools illustrated in Section C.3. For each parameter $p \in \{\alpha, \beta, \delta\}$ we also give the averaged value \bar{p} of the estimates on a set of R realizations of the model and the related mean squared error $MSE(p)$. The detailed procedure works as follows: starting from the estimated values $\hat{\alpha}, \hat{\beta}, \hat{\delta}$ and the observed chosen weights, we generate a sample of R simulated actions-features matrices and we estimate again the parameters on each realization, obtaining the values $\hat{\alpha}_r, \hat{\beta}_r, \hat{\delta}_r$ for $r = 1, \dots, R$. Then we

compute, for each parameter $p \in \{\alpha, \beta, \delta\}$, the average estimate \bar{p} over all the simulations and the $MSE(p)$, as follows

$$\bar{p} = \frac{1}{R} \sum_{r=1}^R \hat{p}_r \quad MSE(p) = \frac{1}{R} \sum_{r=1}^R (\hat{p}_r - \bar{p})^2. \quad (4.6)$$

4.4.2 Check of the asymptotic behaviors

We consider the behavior of the total number of observed features L_t along the time-steps t and we compare it with the theoretical quantity of the model (see Subsection C.1 for its derivation). In particular, for each application we verify that the power-law exponent matches the estimated parameter β . Moreover, we consider the behavior of the total number E_t of edges in the real actions-features matrix and we compare it with the mean number $\langle E_t \rangle$ of edges obtained averaging over R simulated actions-features matrices.

4.4.3 Comparison between real and simulated network and relevance of the weights

We compare the real and simulated network on the basis of the following indicators:

$$\begin{aligned} L_T &= \text{total number of features exhibited by the observed } T \text{ actions,} \\ \bar{O}_T &= \frac{1}{(T-1)} \sum_{t=2}^T O_t \quad \text{with } O_t = \sum_{c=1}^{L_t-1} m_{t,c} \\ \bar{N}_T &= \frac{1}{T} \sum_{t=1}^T N_t. \end{aligned} \quad (4.7)$$

For each action t , with $2 \leq t \leq T$, the quantity O_t is the number of old features shown by action t and $N_t = L_t - L_{t-1}$ is the number of new features brought by action t . The indicators \bar{O}_T and \bar{N}_T provide the averaged values overall the set of observed actions. These indicators are computed for the real network, for the simulated network by the model

described in Section 4.2 with the chosen weights and, in order to evaluate the relevance of the weights inside the dynamics, we also compute them considering all weights equal to 1. In particular, for the simulated matrices, the provided values are an average on R realizations. Essentially, for each indicator $I \in \{L_T, \overline{O}_T, \overline{N}_T\}$ the tables provide the average value

$$\bar{I} = \frac{1}{R} \sum_{r=1}^R I_r$$

where the term I_r denotes the quantity I computed on the r -th simulation of the model. Finally, in order to properly compare the observed values with the model's estimates, we also provide the standard deviation of the estimates, again over a set of R simulations, i.e. the square root of the following quantity

$$\sigma_I^2 = \frac{1}{R} \sum_{r=1}^R (I_r - \bar{I})^2$$

Furthermore, in order to take into account also the not-exhibited “old” features (i.e. the zeros in the matrix \mathbf{M}), we check also the number of “correspondences”, that is we compute the following indicators:

$$\bar{m}_1 = \frac{1}{R} \sum_{r=1}^R m_1^{sim_r} \quad \text{and} \quad \bar{m}_2 = \frac{1}{R} \sum_{r=1}^R m_2^{sim_r}, \quad (4.8)$$

where

$$m_1^{sim_r} = \frac{1}{T-1} \sum_{t=2}^T m_1^{sim_r}(t) \quad \text{with}$$

$$m_1^{sim_r}(t) = \frac{1}{\min(L_{t-1}^{re}, L_{t-1}^{sim_r}, c^*)} \sum_{c=1}^{\min(L_{t-1}^{re}, L_{t-1}^{sim_r}, c^*)} \mathbb{I}_{\{m_{t,c}^{re} = m_{t,c}^{sim_r}\}}$$

and

$$m_2^{sim_r} = \frac{1}{T-1} \sum_{t=2}^T m_2^{sim_r}(t) \quad \text{with}$$

$$m_2^{sim_r}(t) = \frac{|L_t^{re} - L_t^{sim_r}|}{L_t^{re}}.$$

In the above formulas, we use the apex abbreviation re or sim_r to indicate whether the considered quantity is related to the real network or the r -th realization of the simulated network, respectively. The meaning of the above indicators is the following. Given a realization r of the simulated network, for a certain action t , the quantity $m_1^{sim_r}(t)$ calculates the total number of correctly attributed old features among the features in $\{1, \dots, c^*\}$, while $m_2^{sim_r}(t)$ computes the relative error in the total number of observed features. Then, $m_1^{sim_r}$ and $m_2^{sim_r}$ are the corresponding averaged values overall the set of observed actions, and \bar{m}_1 and \bar{m}_2 are the averaged values over the R realizations of the simulated network. Values of \bar{m}_1 and \bar{m}_2 respectively close to 1 and 0 indicate that a very high fraction of features has been correctly allocated by our model and that the relative error in the total number of observed features is very low.

4.4.4 Predictive power of the model

We perform a prediction analysis on the actions-features matrix. More precisely, once a time-step $t^* < T$ is fixed, we estimate the model parameters on the training set corresponding to the set of actions observed at $t = 1, \dots, t^*$. We then employ those estimates to simulate the dynamics of the actions-features network related to the remaining set of actions at times $t = t^* + 1, \dots, T$. Finally, taking the features really observed for these last actions as test set, we evaluate the goodness of our predictions by computing the following indicators:

$$\bar{m}_1^* = \frac{1}{R} \sum_{r=1}^R m_1^{*,sim_r} \quad \text{and} \quad \bar{m}_2^* = \frac{1}{R} \sum_{r=1}^R m_2^{*,sim_r}, \quad (4.9)$$

where

$$m_1^{*,sim_r} = \frac{1}{T - t^*} \sum_{t=t^*+1}^T m_1^{*,sim_r}(t) \quad \text{with}$$

$$m_1^{*,sim_r}(t) = \frac{1}{\min(L_{t-1}^{re}, L_{t-1}^{sim_r}, c^*)} \sum_{c=1}^{\min(L_{t-1}^{re}, L_{t-1}^{sim_r}, c^*)} \mathbb{I}_{\{m_{t,c}^{re} = m_{t,c}^{sim_r}\}}$$

and

$$m_2^{*,sim_r} = \frac{1}{T - t^*} \sum_{t=t^*+1}^T m_2^{*,sim_r}(t) \quad \text{with}$$

$$m_2^{*,sim_r}(t) = \frac{|L_t^{re} - L_t^{sim_r}|}{L_t^{re}}.$$

The interpretation of these quantities is similar to the one proposed in the previous section. Given a realization r of the simulated network, for a certain action t , with $t^* + 1 \leq t \leq T$, the quantity $m_1^{*,sim_r}(t)$ calculates the total number of correctly attributed old features among the features in $\{1, \dots, c^*\}$, while $m_2^{*,sim_r}(t)$ computes the relative error in the total number of observed features. Then, m_1^{*,sim_r} and m_2^{*,sim_r} are the corresponding averaged values over the test set of actions, and \bar{m}_1^* and \bar{m}_2^* are the averaged values over the R realizations of the simulated network. Values of \bar{m}_1^* and \bar{m}_2^* respectively close to 1 and 0 indicate that, starting from the observation of the first t^* actions, a very high fraction of features has been correctly predicted by our model and that the relative error in the total number of observed features is very low.

Regarding the prediction, it is worthwhile to note that if the weights chosen in the model do not depend on t , then it is not necessary to know the agents performing the actions $t^* + 1, \dots, T$, but it is enough to have complete information about the actions in the training set $1, \dots, t^*$. Otherwise, if the weights depend on t , we need to assume also the knowledge of all the agents performing the actions at time-steps $t^* + 1, \dots, T$, in order to take the right weights in the simulation of the model at each time-step $t = t^* + 1, \dots, T$ and predict the corresponding features.

4.5 Datasets

In the present work we employ three different datasets. This section briefly describes each dataset and how it has been obtained. The complete description of the data cleaning process is provided in Appendix C.

- **IEEE dataset for Automatic Driving.** For our first application we have downloaded (on June 26, 2018) all papers published between 2000 and 2003 present in the IEEE database [144] in the scientific research field of Automatic Driving. We have performed the research following the same criteria as in [127], selecting all papers containing at least one of the keywords: Lane Departure Warning, Lane Keeping Assist, Blindspot Detection, Rear Collision Warning, Front Distance Warning, Autonomous Emergency Braking, Pedestrian Detection, Traffic Jam Assist, Adaptive Cruise Control, Automatic Lane Change, Traffic Sign Recognition, SemiAutonomous Parking, Remote Parking, Driver Distraction Monitor, V2V or V2I or V2X, Co-Operative Driving, Telematics & Vehicles, and Night vision. For each paper we have at our disposal all the bibliographic records, such as title, full abstract, authors' names, keywords, year of publication, date in which the paper was added to the IEEE database, and many others.
- **ArXiv dataset for Theoretical High Energy Physics.** Our second application of the model has been performed with the arXiv dataset of publications in the scientific area of Theoretical High Energy Physics (Hep-Th), freely available from [131, 143]. The dataset collects a sample of text files reporting the full frontispiece of each paper, so we have information on: arXiv id number, submission date, name and email of the author who made the submission, title, authors' names and the entire text of the abstract.
- **Instagram dataset.** This dataset has been kindly provided by Prof. Emilio Ferrara [145] and a more detailed description of it can be found in [132]. Very briefly, the dataset has been crawled through the Instagram API between January 20 and February 17, 2014 and

collects public media (with their author, timestamp and set of hash-tags) as well as users information (with their list of followers and followees) of a set of 2100 anonymized participants to 72 popular photographic contests that took place between October 2010 and February 2014. The overall media dataset records more than one million posts but, with the purpose of maximizing the density of our actions-features matrix, we considered only those posts published during the weekends in the crawling period (Jan 20 to Feb 17, 2014) in which at least five hashtags are used.

4.6 Results

In the present section we provide the results of our analysis, that consists in the application of the method to each dataset.

4.6.1 IEEE dataset for Automatic Driving

For our first application we have used the IEEE dataset for Automatic Driving described in the previous section. For each paper we have at our disposal all the bibliographic records, such as title, full abstract, authors' names, keywords, year of publication, date in which the paper was added to the IEEE database, and many others. The papers have been sorted chronologically according to the date in which they were added to the database. We have considered all nouns and adjectives (from now on "key-words") included in the title or abstract as the features of our model and sorted them according to their arrival time. See Appendix C for a more detailed description of the data preparation procedure. The features matrix obtained at the end of the cleaning procedure collects $T = 492$ papers (actions) recorded in the period 2000 – 2003 and involving $n = 1251$ distinct authors (agents) and containing $L_T = 4553$ key-words (features). The binary entry $m_{t,c}$ of the actions-features matrix indicates whether feature c is present into the title or the abstract of the paper recorded at time-step t . A pictorial representation of the matrix is provided in Figure 29.

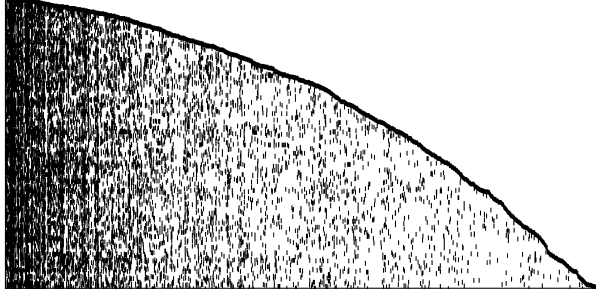


Figure 29: IEEE Automatic Driving. Observed actions-features matrix with dimensions $T \times L_T = 492 \times 4553$. Black dots represent 1 while white dots represent 0.

p	\hat{p}	\bar{p}	$MSE(p)$
α	68.534	68.699	14.699
β	0.5962	0.5963	0.0001
δ with $x_{i,t}^{pub}$	$\approx 2.22 \cdot 10^{-16}$	$\approx 4.36 \cdot 10^{-5}$	$\approx 5.90 \cdot 10^{-9}$
δ with $x_{i,t}^{col}$	$\approx 2.26 \cdot 10^{-16}$	$\approx 5.01 \cdot 10^{-5}$	$\approx 6.82 \cdot 10^{-9}$

Table 3: IEEE Automatic Driving. Estimation of the model parameters. The average values \bar{p} and the parameters' MSE are computed over $R = 100$ realizations of the model. See Section 4.4 for further details.

For this application, we use weights of the third type in Section 4.3.2. Indeed, at each time-step t , we associate to each author i a “fitness” variable $x_{i,t}$ that quantifies the influence of author i in the considered research field, and we define the weights as

$$w_{t,j,c} = w_{t,j} = e^{-1/y_{t,j}} \text{ with } y_{t,j} = \max\{x_{i,t} : i \in \mathcal{J}(j)\} \text{ where } \mathcal{J}(j) = \text{set of the agents performing action } j. \quad (4.10)$$

Therefore the inclusion probability in equation (4.3) reads as

$$p_{t,c} = \frac{\delta}{2} + (1 - \delta) \frac{\sum_{j=1}^{t-1} m_{j,c} e^{-1/y_{t,j}}}{t}. \quad (4.11)$$

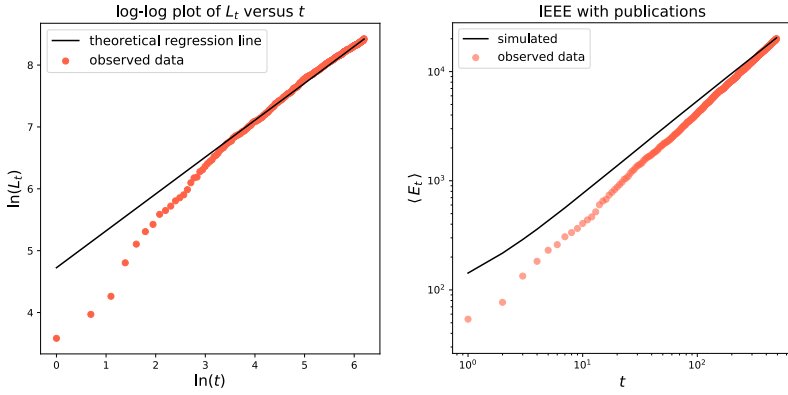


Figure 30: IEEE Automatic Driving. Left: Plot of $\ln(L_t)$ as a function of $\ln(t)$, with the power-law trend. The red dots refer to the real data and the black line gives the theoretical regression line with slope $\hat{\beta}$. Right: Asymptotic behavior of the number of edges in the actions-features network. Red dots refer to E_t of the real data, while the black line shows $\langle E_t \rangle$ obtained by the model with $x_{i,t}^{pub}$ (averaging over $R = 100$ simulations).

The term $y_{t,j}$ is the maximum among the fitness variables $x_{i,t}$ at time-step t of the authors $i \in \mathcal{J}(j)$, i.e. those who published the paper appeared at time-step j . A high value of $x_{i,t}$ should identify a person who is relevant in the considered research field so that it is likely that other scholars use the same features of her/his actions, that essentially are the keywords related to her/his research. As a consequence, in the preferential attachment term, we give to each old feature c a weight that is increasing with respect to the fitness variables of the authors who included c in their papers. We analyse two different fitness variables:

$$x_{i,t}^{pub} = (\text{number of author } i\text{'s publications until time } t - 1) + 1 \quad (4.12)$$

and

$$x_{i,t}^{col} = (\text{number of author } i\text{'s collaborators until time } t - 1) + 1. \quad (4.13)$$

Note that we count the publications or collaborators until time-step $t - 1$, that is until a time-step before the publication time of paper t and 1 is added in order to avoid division by zero in the previous formula (4.10). The performed analysis follows the outline explained in Section 4.4: we first estimate the model's parameters, obtaining the results in Table 3. According to it, the weighted preferential attachment term (4.5) plays a predominant role, due to the estimated value obtained for the parameter δ that is approximately zero. Figure 30 provides in the left panel a log-log plot of the cumulative count of new features (key-words) as a function of time (see the red dots) that clearly shows a power-law behavior. This agrees with the theoretical property of the model stated in Appendix C, Subsection C.1, according to which the power-law exponent has to be equal to the parameter β . This fact is checked in the plot by the black line, which slope is the estimated value of the parameter β .

The goodness of fit of our model to the dataset has been evaluated through the computation of the quantities (4.7) and (4.8). These results are shown in Tables 4 and 5. Table 4 shows that our model reproduces the total number L_T of features observed at the end of the observation period T , as well as the average number of new features \overline{N}_T in all the three considered cases. The average number of old features (i.e. the quantity

Matrix	L_T	σ_{L_T}	\overline{O}_T	σ_{O_T}	\overline{N}_T	σ_{N_T}
real	4553		31.54		9.25	
weights with $x_{i,t}^{pub}$	4558	68.19	32.13	1.29	9.26	0.14
weights with $x_{i,t}^{col}$	4548	63.00	54.20	2.88	9.25	0.13
weights = 1	4551	71.50	134.7	3.48	9.25	0.15

Table 4: IEEE Automatic Driving. Comparison between real and simulated networks by means of the indicators (4.7). We also include an estimate of the variations around the average values, through the computation of the sample standard deviations. See Section 4.4 for further details.

weights with $x_{i,t}^{pub}$	\overline{m}_1	\overline{m}_2
$c^* = 4553$ (all observed features)	0.97	0.047
$c^* = 100$	0.88	
$c^* = 200$	0.90	
$c^* = 300$	0.91	
weights with $x_{i,t}^{col}$	\overline{m}_1	\overline{m}_2
$c^* = 4553$ (all observed features)	0.96	0.049
$c^* = 100$	0.83	
$c^* = 200$	0.86	
$c^* = 300$	0.88	
weights = 1	\overline{m}_1	\overline{m}_2
$c^* = 4553$ (all observed features)	0.93	0.049
$c^* = 100$	0.55	
$c^* = 200$	0.64	
$c^* = 300$	0.70	

Table 5: IEEE Automatic Driving. Comparison between real and simulated matrices by means of the indicators (4.8). The first row of each table evaluates the indicators on the whole matrix ($c^* = 4553$), while the other rows show the results computing the indicators only on the first c^* ($= 100, 200, 300$) features.

weights with $x_{i,t}^{pub}$	\overline{m}_1^*	\overline{m}_2^*
$t^* = 369$ and $c^* = 4553$	0.99	0.017
$t^* = 246$ and $c^* = 4553$	0.98	0.060
$t^* = 123$ and $c^* = 4553$	0.98	0.114
$t^* = 369$ and $c^* = 200$	0.93	
$t^* = 246$ and $c^* = 200$	0.93	
$t^* = 123$ and $c^* = 200$	0.93	

Table 6: IEEE Automatic Driving. Predictions on the actions-features matrix. The indicators (4.9) are computed for different levels of information used as “training set”: more precisely, the different t^* correspond to 75%, 50% and 25% of the set of the actions, respectively. Moreover, the indicators are computed on the whole matrix ($c^* = 4553$) and also taking into account only the first $c^* = 200$ features.

\overline{O}_T) is well reproduced only in the case with $x_{i,t}^{pub}$ (that is the case with the fitness based on the number of publications). Table 5 also indicates that the model with $x_{i,t}^{pub}$, or with $x_{i,t}^{col}$, shows a better performance than the one with all the weights equal to one. More precisely, the values obtained for the indicator \overline{m}_2 are almost the same for all the three cases (the average error on the total number of arrived features is around 5%); while the most significant differences are in the values of the indicator \overline{m}_1 . Indeed, for the model with the fitness $x_{i,t}^{pub}$, the computed value of \overline{m}_1 ranges from 88% to 97%, pointing out that a high percentage of the entries in the actions-features matrix have been correctly inferred by the model. The same value for the model with the fitness $x_{i,t}^{col}$ ranges from 83% to 96%, and, for the model with all the weights equal to 1, it ranges from 55% to 93%. The differences with respect to the case with all the weights equal to 1 are more evident when we select the first c^* features: indeed, with $x_{i,t}^{pub}$ we succeed to infer the value of at least 88% of the entries; while with all the weights equal to 1 the percentage remains under 70%. This means that the major difference in the performance of the different considered weights is in the first features, that are those for which the preferential attachment term is more relevant, since they appeared in the system at an earlier stage. At this point, we choose the model that

takes into account the authors' number of publications as the best performing one for the considered dataset and in the following analysis we focus on it.

In Table 6 we evaluate the predictive power of the model: we estimate the parameters of the model only on a subset of the observed actions, respectively the 75%, 50% and 25% of the total observations; we then predict the features for the "future" actions $\{t^* + 1, \dots, T\}$ and compare the predicted and observed results by means of the indicators in (4.9) over the whole set of features and only on a portion of it. The indicator \overline{m}_1^* ranges from 93% to 99%. Finally, in the right panel of Figure 30, we provide the asymptotic behavior of the number of edges in the actions-features network: more precisely, the red dots represent the total number E_t of edges observed in the real actions-features matrix at each time-step; while the continuous black line shows the mean number $\langle E_t \rangle$ of edges obtained averaging over $R = 100$ simulations of the model with the chosen weights.

It is worth noting that the difference in performance between the two definitions of fitness variables has a straightforward interpretation: in the considered case, i.e. for the publications in the area of Automatic Driving in the considered period, the relevance of an author (with respect to the probability of transmitting her/his features) is better measured by considering the number of her/his publications rather than the number of her/his co-authors. As we will see later on, we get a different result for our second application.

4.6.2 ArXiv dataset for Theoretical High Energy Physics

Our second application of the model has been performed with the arXiv dataset of publications in the scientific area of Theoretical High Energy Physics (Hep-Th) [131]. From the original format we isolate the submission date and the identity number of the paper, in order to sort all papers (actions) chronologically. Then, with the final purpose of constructing the features matrix, we consider all key-words included either in the main title or in the abstract as the features of the papers and we sort

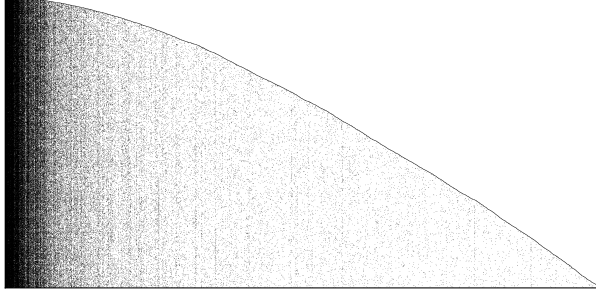


Figure 31: arXiv High Energy Physics. Observed actions-features matrix with dimensions $T \times L_T = 10603 \times 22304$. Black dots represent 1 while white dots represent 0.

p	\hat{p}	\bar{p}	$MSE(p)$
α	40.812	40.940	2.4816
β	0.6305	0.6303	$2.34 \cdot 10^{-5}$
δ with $x_{i,t}^{pub}$	$\approx 2.22 \cdot 10^{-16}$	$\approx 1.41 \cdot 10^{-6}$	$\approx 3.56 \cdot 10^{-12}$
δ with $x_{i,t}^{col}$	$\approx 2.22 \cdot 10^{-16}$	$\approx 1.06 \cdot 10^{-5}$	$\approx 2.31 \cdot 10^{-12}$

Table 7: arXiv High Energy Physics. Estimates of the model parameters. The average values \bar{p} and the parameters' MSE are computed over $R = 100$ realizations of the model. See Section 4.4 for further details.

them according to their time of appearance. The complete data preparation phase is described in Appendix C. Finally, we construct the features matrix \mathbf{M} , which elements $m_{t,c} = 1$ if paper t includes key-word c either in the title or in the abstract and $m_{t,c} = 0$ otherwise. For this phase we have considered all papers published from 2000 to 2003 and included in the repository. The result is shown in Figure 31, where the observed actions-features matrix collects $T = 10603$ papers (actions) registered between 2000 and 2003 and $L_T = 22304$ key-words appeared in the title or in the abstract (features), while the total number of involved authors (agents) is $n = 5633$.

The weights for this application are defined as in the previous one,

Matrix	L_T	σ_{L_T}	\overline{O}_T	σ_{O_T}	\overline{N}_T	σ_{N_T}
real	22304		28.42		2.10	
weights with $x_{i,t}^{pub}$	22305	145.83	15.37	0.41	2.10	0.02
weights with $x_{i,t}^{col}$	22317	148.35	18.79	0.70	2.10	0.02
weights = 1	22313	147.35	97.16	5.16	2.10	0.02

Table 8: arXiv High Energy Physics. Comparison between real and simulated networks by means of the indicators (4.7). We also include an estimate of the variations around the average values, through the computation of the sample standard deviations. See Section 4.4 for further details.

weights with $x_{i,t}^{pub}$	\overline{m}_1	\overline{m}_2
$c^* = 22304$ (all observed features)	0.995	0.022
$c^* = 5576$	0.992	
$c^* = 11152$	0.994	
$c^* = 16728$	0.995	
weights with $x_{i,t}^{col}$	\overline{m}_1	\overline{m}_2
$c^* = 22304$ (all observed features)	0.995	0.021
$c^* = 5576$	0.991	
$c^* = 11152$	0.994	
$c^* = 16728$	0.994	
weights = 1	\overline{m}_1	\overline{m}_2
$c^* = 22304$ (all observed features)	0.987	0.021
$c^* = 5576$	0.976	
$c^* = 11152$	0.985	
$c^* = 16728$	0.986	

Table 9: arXiv High Energy Physics. Comparison between real and simulated matrices by means of the indicators (4.8). The first row of each table evaluates the indicators on the entire matrix ($c^* = 22304$), while the others show the results computing the indicators only on the first c^* (= 5576, 11152, 16728) features.

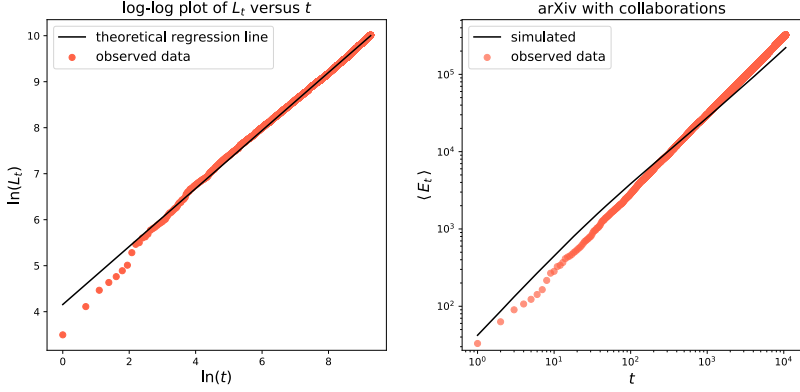


Figure 32: arXiv High Energy Physics. Left: Plot of $\ln(L_t)$ as a function of $\ln(t)$, with the power-law trend. The red dots refer to the real data and the black line gives the theoretical regression line with slope $\hat{\beta}$. Right: Asymptotic behavior of the number of edges in the actions-features network. Red dots refer to E_t of the real data, while the black line shows $\langle E_t \rangle$ obtained by the model with $x_{i,t}^{col}$ (averaging over $R = 100$ simulations).

described in equation (4.10). We consider again the two different definitions for the fitness term $x_{i,t}$ (see equations (4.12) and (4.13)). For both the definitions of fitness, we perform the analysis following the methodology explained in Section 4.4. We first estimate the model's parameters, obtaining the results in Table 7.

We see that the weighted preferential attachment term (4.5) is the major contributor due to the estimated value obtained for the parameter δ that is essentially zero. Figure 32 provides in the left panel a log-log plot of the cumulative count of new features (key-words) as a function of time (see the red dots), that clearly shows a power-law behavior. This agrees with the theoretical property of the model stated in Appendix C according to which the power-law exponent has to be equal to the parameter β . This fact is checked in the plot by the black line, which slope is the estimated value of the parameter β .

The goodness of fit of our model to the dataset has been evaluated

through the computation of the quantities (4.7) and (4.8). These results are shown in Table 8 and Table 9. Table 8 shows that our model is able to reproduce the total number L_T of features observed at the end of the observation period T and the average number of new features \overline{N}_T . Instead, the average number of old features (i.e. the quantity \overline{O}_T) is underestimated by the model with the weights based on $x_{i,t}^{pub}$ and $x_{i,t}^{col}$, while it is widely over-estimated in the case with all the weights equal to 1. The discrepancy in the values is smaller for the case with $x_{i,t}^{col}$ (that is the case with the fitness based on the number of collaborators). Table 9 shows that the performance of the model in reproducing the data are comparable, with both the considered definitions of fitness and they are also good for the case with all weights equal to one.

At this point, we choose the model that takes into account the authors' number of collaborations and the last analysis focuses on it. In Table 10 we evaluate the predictive power of the model: we estimate the parameters only on a subset of the observed actions, respectively the 75%, 50% and 25% of the total observations; we then predict the features for the "future" actions $\{t^* + 1, \dots, T\}$ and compare the predicted and observed results by means of the indicators in (4.9) over the whole set of features and only on a portion of it. In particular, we obtain that the indicator \overline{m}_1^* is almost always equal to 99%. Finally, in the right panel of Figure 32, we provide the asymptotic behavior of the number of edges in the actions-features network: more precisely, the red dots represent the total number E_t of edges observed in the real actions-features matrix at each time-step; while the continuous black line shows the mean number $\langle E_t \rangle$ of edges obtained averaging over $R = 100$ simulations of the model with the chosen weights.

Contrarily to the previous case, in this application we observe a comparable performance of the model with both the considered definitions of fitness. This means that, for the publications in High Energy Physics in the considered period, both the number of co-authors and the number of publications of an author can be considered as reasonable measures in order to evaluate her/his relevance in the research field.

weights with $x_{i,t}^{col}$	\overline{m}_1^*	\overline{m}_2^*
$t^* = 7952$ and $c^* = 22304$	0.997	0.006
$t^* = 5302$ and $c^* = 22304$	0.997	0.026
$t^* = 2651$ and $c^* = 22304$	0.996	0.035
$t^* = 7952$ and $c^* = 11152$	0.996	
$t^* = 5302$ and $c^* = 11152$	0.996	
$t^* = 2651$ and $c^* = 11152$	0.996	

Table 10: arXiv High Energy Physics. Predictions on the actions-features matrix. The indicators (4.9) are computed for different levels of information used as “training set”: more precisely, the different t^* correspond to 75%, 50% and 25% of the set of the actions, respectively. Moreover, the indicators are computed on the whole matrix ($c^* = 22304$) and also taking into account only the first $c^* = 11152$ features.

4.6.3 Instagram dataset

In this section we apply the proposed model to the dataset of the social network of Instagram, presented in [132]. The procedure of posts selection yields a sample of $T = 2151$ posts (actions) and $L_T = 5890$ hashtags (features). The available posts were ordered chronologically according to the associate timestamp of publication and the hashtags (features) were sorted in terms of their first appearance in a post. After this first phase of data arrangement, we construct the actions-features matrix \mathbf{M} , with $m_{t,c} = 1$ if post t contains hashtag c and $m_{t,c} = 0$ otherwise. The resulting matrix is shown in Figure 33, with non-zero values indicated by black points.

For this application, we chose weights that depend on an indicator related to the underlying Instagram network (see Subsection 4.3.2). Precisely, we associate to each agent i the variable x_i defined as the number of agents i ’s followers, among those who were active during the crawling period and we set

$$w_{t,j,c} = w_t = e^{-x_{i(t)}}, \quad (4.14)$$

where $i(t)$ denotes the agent performing action t . Therefore the inclusion

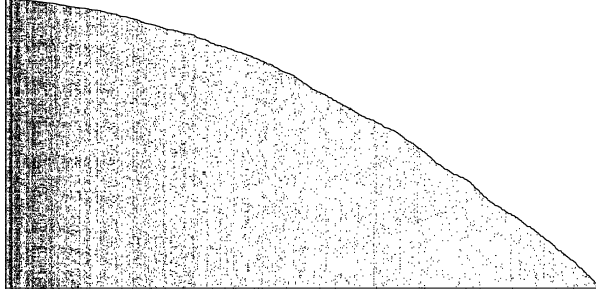


Figure 33: Instagram. Observed actions-features matrix, with dimensions $T \times L_T = 2151 \times 5890$. Black dots represent 1 while white dots represent 0.

probability for hashtag c becomes

$$p_{t,c} = \frac{\delta}{2} + (1 - \delta) \frac{\sum_{j=1}^{t-1} m_{j,c}}{t} e^{-x_{i(t)}}, \quad (4.15)$$

where the average popularity of hashtag c is exponentially discounted by the factor $x_{i(t)}$. The decision to introduce such kind of weights was driven by the following consideration. A user with a very high number of followers identifies a person who is very popular on the social networks, an “influencer” in the extreme case. As a consequence, it may be reasonable to think that she/he is less affected by other people’s posts and, consequently, less prone to use “old” hashtags. For this user, the average popularity of c in the inclusion probability $p_{t,c}$ should be less relevant. On the contrary, a user with a low number of followers may be more incline to follow the current trends and the others’ preferences and choices. It is worthwhile to point out that in the definition of the weights, we considered the number of followers of an user as fixed to the value we observed at the end of the period of observation. In general, it may change in time, depending on the changes in her/his network of “virtual friendships”. However, we assume it to be constant because of the short time span considered.

The performed analysis follows the outline explained in Section 4.4 (for the simulated matrices, all the considered quantities have been av-

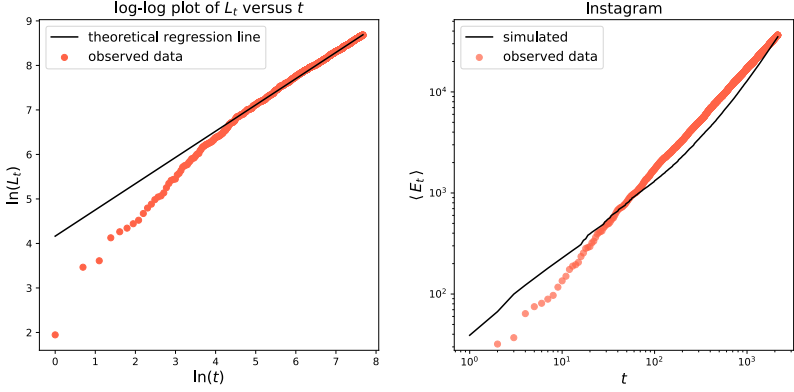


Figure 34: Instagram. Left: Plot of $\ln(L_t)$ as a function of $\ln(t)$, with the power-law trend. The red dots refer to the real data and the black line gives the theoretical regression line with slope $\hat{\beta}$. Right: Asymptotic behavior of the number of edges in the actions-features network. Red dots refer to E_t of the real data, while the black line shows $\langle E_t \rangle$ obtained by the model (averaging over $R = 100$ simulations).

eraged over $R = 100$ realizations of the model). We first estimate the model's parameters, obtaining the results in Table 11. The weighted preferential attachment term (4.5) plays again an important role, but slightly lower than in the previous cases, since the inclusion probability is obtained with $\delta = 0.6\%$. Figure 34 provides in the left panel a log-log plot of the cumulative count of new features (hashtags) as a function of time (see the red dots), that clearly shows a power-law behavior. This agrees with the theoretical property of the model stated in Appendix C, according to which the power-law exponent has to be equal to the parameter β . This fact is checked in the plot by the black line, whose slope is the estimated value of the parameter β .

The goodness of fit of our model to the dataset has been evaluated through the computation of the quantities (4.7) and (4.8). These results are shown in Table 12 and Table 13. Table 12 shows that our model is perfectly able to reproduce the total number L_T of features observed at the

p	\hat{p}	\bar{p}	$MSE(p)$
α	37.895	37.843	4.516
β	0.5897	0.5900	$7.89 \cdot 10^{-5}$
δ with weights (4.14)	0.0063	0.0062	$2.70 \cdot 10^{-8}$

Table 11: Instagram: Estimation of the model parameters. The average values \bar{p} and the parameters' MSE are computed over $R = 100$ realizations of the model. See Section 4.4 for further details.

Matrix (with $T = 2151$)	L_T	σ_{L_T}	\bar{O}_T	σ_{O_T}	\bar{N}_T	σ_{N_T}
real	5890		14.23		2.74	
weights (4.14)	5885	42.39	13.56	0.24	2.74	0.02
weights = 1	5899	73.00	80.03	5.20	2.74	0.04

Table 12: Instagram. Comparison between real and simulated networks by means of the indicators (4.7). We also include an estimate of the variations around the average values through the computation of the sample standard deviations. See Section 4.4 for further details.

end of the observation period T , as well as the average number of new features \bar{N}_T in both the considered cases. The average number of old features (i.e. the quantity \bar{O}_T) shows a good agreement with the observed quantity in the case of the model with the chosen weights, contrarily to the model with all the weights equal to one for which we obtain a much higher value. Also Table 13 indicates that the model with the chosen weights shows a better performance than the one with all the weights equal to one. More precisely, the values obtained for the indicator \bar{m}_2 are almost the same for both cases (the average error on the total number of arrived features is around 4%); while the most significant differences are in the values of the indicator \bar{m}_1 . Indeed, for the model with the chosen weights, the computed values of \bar{m}_1 range from 97% to 99%, pointing out that a high percentage of the entries in the actions-features matrix have been correctly inferred by the model. The differences are more evident when we select the first c^* features: indeed, with the chosen weights we succeed to infer the values of at least 97% of the entries; while with all the weights equal to 1 the percentage remains under 86%. This means

weights (4.14)	\overline{m}_1	\overline{m}_2
$c^* = 5890$ (all observed features)	0.99	0.037
$c^* = 100$	0.97	
$c^* = 250$	0.98	
$c^* = 500$	0.98	
weights = 1	\overline{m}_1	\overline{m}_2
$c^* = 5890$ (all observed features)	0.97	0.037
$c^* = 100$	0.63	
$c^* = 250$	0.77	
$c^* = 500$	0.86	

Table 13: Instagram. Comparison between real and simulated matrices by means of the indicators (4.8). The first row of each table evaluates the indicators on the whole matrix ($c^* = 5890$), while the other rows show the results computing the indicators only on the first c^* ($= 100, 250, 500$) features.

that the major difference in the performance of the different considered weights is in the first features, that are those for which the preferential attachment term is more relevant.

In Table 14 we evaluate the predictive power of the model with the chosen weights: we estimate the parameters of the model only on a subset of the observed actions, respectively the 75%, 50% and 25% of the total observations; we then predict the features for the “future” actions $\{t^* + 1, \dots, T\}$ and compare the predicted and observed results by means of the indicators in (4.9) over the whole set of features and only on a portion of it. The indicator \overline{m}_1^* ranges from 97% to 99%. Finally, in the right panel of Figure 34, we provide the asymptotic behavior of the number of edges in the actions-features network: more precisely, the red dots represent the total number E_t of edges observed in the real actions-features matrix at each time-step; while the continuous black line shows the mean number $\langle E_t \rangle$ of edges obtained averaging over $R = 100$ simulations of the model with the chosen weights.

weights (4.14)	\overline{m}_1^*	\overline{m}_2^*
$t^* = 1613$ and $c^* = 5890$	0.99	0.006
$t^* = 1076$ and $c^* = 5890$	0.99	0.031
$t^* = 538$ and $c^* = 5890$	0.99	0.099
$t^* = 1613$ and $c^* = 250$	0.98	
$t^* = 1076$ and $c^* = 250$	0.98	
$t^* = 538$ and $c^* = 250$	0.97	

Table 14: Instagram. Predictions on the actions-features matrix. The indicators (4.9) are computed for different levels of information used as “training set”: more precisely, the different t^* correspond to 75%, 50% and 25% of the set of the actions, respectively. Moreover, the indicators are computed on the whole matrix ($c^* = 5890$) and also taking into account only the first $c^* = 250$ features.

4.6.4 Summary of the results

In all the three cases we selected the weights depending on a fitness variable. In the first two applications (IEEE and arXiv), the fitness variable measures the “ability” of the agents (authors) to transmit the features (keywords) of their actions (publications). In the third application (Instagram) the fitness variable quantifies the “inclination” of the agents (users) to follow the features (hashtags) of the previous actions (posts). From the performed analyses of the actions-features bipartite networks, we get the following main common issues for the three applications:

- The preferential attachment rule plays a relevant role in the formation of the actions-features network, because of the small estimated values obtained for the parameter δ . In particular, in the first two applications, the estimated value of δ is very close to zero.
- The considered indicators (4.7), (4.8) and (4.9), and the plots regarding the behavior in time of the total number of observed features L_t and the total number of edges E_t show a very good fit between the model with the chosen weights and the real datasets. In particular, the power-law behavior of L_t perfectly matches the theoretical one with the estimated parameter β as the power-law exponent, and a

high percentage of the entries of the actions-features matrix is successfully inferred with the model. Moreover, a good performance is also obtained when making a prediction analysis, i.e. testing the percentage of the entries that are successfully recovered by the model providing it with different levels of information.

- With respect to the “flat weights”, i.e. all weights equal to 1, the chosen weights guarantee a much better agreement with the real actions-features matrices. Among the indicators in (4.7), the one that mostly highlights this is \overline{O}_T . Moreover, the difference in the performance of the model with different weights is also particularly evident when we consider a subset of the overall set of observed features for the computation of the indicators \overline{m}_1 in (4.8) and \overline{m}_1^* in (4.9). Indeed, the first features are those for which the preferential attachment term is more relevant.

4.7 Discussion and conclusions

In this work we have presented our contribution to the stream of literature regarding stochastic models for bipartite networks formation. With respect to the previous publications, our paper introduces some novelties. First of all, given a system of agents, we are not interested in modeling the process of link formation between the agents themselves, we instead define a model that describes the activity of the agents, studying the behavior in time of agents’ actions and the features shown by these actions. This issue allows to amplify the range of possible applications, since we only assume to know the chronological order in which we observe the agents’ actions, and not the order in which the agents arrive. Second, we extend the concept of “preferential attachment with weights” [29, 30] to this framework. The weights can have different shapes and meaning according to the specific setting considered and play an important role since the probability that a future action shows a certain feature depends, not only on its “popularity” (i.e. the number of previous actions showing the feature) as stated by the preferential attachment rule,

but also on some characteristics of the agents and/or the features themselves. For instance, the weights may provide information regarding the “ability” of an agent to transmit the features of her/his actions to the future actions, or the “inclination” of an agent to adopt the features shown in the past.

Summarizing, we first provide a full description of the model dynamics and interpretation of the included parameters and variables, also showing some theoretical results regarding the asymptotic properties of some important quantities. Moreover, we illustrate the necessary tools in order to estimate the parameters of the model and we consider three different applications. For each of them, we evaluate the goodness of fit of the model to the data by checking the theoretical asymptotic properties of the model in the real data, by comparing several indicators computed both on the real and simulated matrices, as well as testing the ability of the model as a predictive instrument in order to forecast which features will be shown by future actions. All our analyses point out a very good fit of our model and a very good performance of the adopted tools in all the three considered cases.

Our model and the related analysis have been able to detect some interesting aspects that characterize the different examined contexts. In the first two applications (IEEE and arXiv) we examined the publications in the scientific areas of Automatic Driving and of High Energy Physics (briefly Hep-Th) and we took into account two kinds of fitness variables for the authors: one based on the number of publications and the other based on the number of collaborators. Our study reveals that, for Hep-Th, both the number of publications of an author and the number of her/his collaborators are able to provide a good agreement with real data, while for Automatic Driving we found a better performance of the model with the weights based on the number of publications. Probably this difference is due to the fact that, while the Physics of High Energies is quite an old subject in which different branches developed, Automatic Driving is a much more recent, and so limited, research area. See for example the observed values of T and L_T , the number of publications and the number of keywords in the considered period, that are much

smaller for Automatic Driving than the ones observed for Hep-Th. The behavior of on-line social networks is completely different: we examine the dataset of Instagram, with posts considered as actions and hashtag as features. We observed that the less followers a user has the higher the number of “old” hashtag used. This could be related to the fact that less popular users tend to re-use many “old” hashtags in order to increase their visibility, while highly famous users do not feel the need of improving their popularity and focus on few “old” hashtags only. Indeed, this behavior shows a completely different role of the “on-line followership” relations respect to coauthorships: while collaborations incentive the usage of a high number of existing features, the number of followers takes to a limited usage of existing hashtags.

Chapter 5

Conclusions

This work explores some of the many facets of social network analysis: each chapter takes into account a different type of social network and is based on appropriate tools to analyse it. We here summarize the main topics this work has dealt with and the main findings of the manuscript.

Chapter 2 is focused on the analysis of *rating networks*. These graphs are essentially bipartite networks of reviews, which nodes represent users and the products they purchase on e-commerce websites, while the edges are weighted by the numerical score assigned to the purchase. We first provide a formal definition of these systems in terms of a network structure, then we deal with the problem of constructing a suitable null model for their analysis. At first we extended the frameworks of the Weighted Configuration Model and Erdős-Rényi Random Graph to the case of networks with a finite sequence of integer weights (i.e. the case considered here). However, these tools turned out to be unsuitable to represent the main topological quantities of these systems. Therefore, we proceeded with the extension of the Configuration Model framework to the specific case of rating networks, introducing the entire distribution of scores received by a node as constraint of the problem. Essentially, the main novelty of this approach is that we do not simply specify the overall score received by a node but we constrain over the specific number of reviews with a given rating. The introduction of this enlarged pool of constraints

allows to build a more restrictive null model that is able to reproduce the main topological quantities better than the analysed alternatives. Moreover, our method recovers the probability that a user positively or negatively reviews a specific product in an unbiased way and without any a-priori information regarding users' habits. Therefore, this information may be taken into account in future developments of recommendation systems, in order to account for network topology and discount for the tendency of a user to give a specific score-review to an item.

Chapter 3 instead deals with the topic of *online social networks*. We have collected a dataset regarding the tweets posted during the last Italian elections that took place in March 2018. At first, since this kind of data is not labeled (and a manual classification procedure would lead to a very expensive work), we have exploited the way people consume the news in order to identify groups of connected verified accounts. With this procedure we managed to classify the certified users in different coalitions, that mostly overlap with the observed division in political parties. This first kind of analysis highlights that, despite the data-driven approach used to label the data, the users' behaviour on the social networks provides some insights regarding one's political opinion and point of view. The same division has been extended also to non-verified accounts by means of a polarization index that we define, that essentially counts the amount of interactions towards each community. In this case our analysis confirms the results already present in the literature regarding this topic: users in online social network contexts tend to share contents that reflect their pre-existing ideology while ignoring the conflicting ones [89, 90, 91, 92, 93]. Finally, we propose a representation of the tweeting and retweeting activity as a bipartite and directed network, that collects users on one layer and tweets on the opposite one. In this graph, an outgoing edge connecting a user to a tweet indicates that the account has tweeted the post for the first time; a link in the opposite direction, connecting instead a tweet to a user, indicates that the post has been retweeted by the account at least once. A null model for this kind of networks has been already presented in the literature by [104]; we briefly review the methodology and then we employ this ensemble construction

method to extend the projection and validation procedure presented in [38] to the directed case. By doing so, in the resulting monopartite and directed network of users we manage to identify the most prominent content-sharers, that have been mostly retweeted during the observation period. The obtained network reveals interesting topological structures internal to the political coalitions, that also provide some insights regarding future political alliances: indeed we detect a significant flow of tweets between the Lega and M5S, the two political parties that built a coalition *after* the election, to form a new government.

Finally, Chapter 4 analyses the two cases of *collaboration networks* [10] and *online social networks*. More specifically, it presents a stochastic model that reproduces the activity of a set of users in a generic social network. This element per se represents a novelty in this literature, since we are not interested in modeling the formation of links among the users; on the contrary, we only describe the evolution of the system involving users' actions and the features shown by these actions. Therefore the model does not require to know the order in which the actors appear, that in many situations could be unknown or not relevant, but solely specifies the order in which the actions are performed. We introduce a rule called "preferential attachment with weights" that partially regulates the formation of links in the bipartite actions-features network: each action can independently show a set of new features (not present in the system when the action is performed) or adopt one of the already-existing features. The choice of the old features is not only driven by the popularity of such characteristics (i.e. the number of previous actions showing them, that is the classical preferential attachment mechanism) but is also affected by certain quantities, called "weights", that take into account some personal traits of the actions' performers. These weights have various interpretations according to the considered settings: we provide some possible examples, together with their interpretation. Moreover, we show some theoretical properties of the model and introduce statistical tools for the parameters' estimation. Finally we apply the model to three different datasets: two of them describe scientific collaboration in different research fields (the arXiv dataset for Theoretical High Energy

Physics and the IEEE dataset for Automatic Driving), the third one instead collects posts published on the online social network of Instagram. For each case we specify the employed definition of weights and provide a detailed description of the obtained results and their discussion.

In summary, this thesis deals with a very popular topic nowadays that is the study of social networks: the approaches implemented for their analysis are substantially different but equally relevant in their findings to describe the systems in question. In broad terms, one aspect that these chapters highlight is the importance of using benchmarks to extract information from an, otherwise very complicated, system. At first we have provided an entropy-based model useful to describe bipartite rating networks; then the construction of a suitable model for the social network of Twitter manages to identify the significant content spreaders and shed light on the structural skeleton of this architecture. In both cases considered in Chapters 2 and 3, the topology of the network, i.e. the activity of the involved nodes, is solely employed to construct a reference null-model. Therefore we may conclude that any significant behaviour is detected *discounting* for the activity of nodes, *net of* the graph topology. On the contrary the generative model presented in Chapter 4 considers a wider range of elements to describe a system: the graph topology, i.e. the information about nodes' degrees summarized in the preferential attachment rule, is enriched with additional ingredients, the weights, that take into account some aspects of the actions' performers or the actions themselves. In the considered cases we find good agreement in how the model manages to describe and reproduce the system under analysis, therefore we may consider these approaches to network modeling equally valuable, although extremely different in their nature. Another aspect that is worth stressing the attention on is the temporal representation of the network considered in the three chapters. On the one hand, the models introduced in Chapters 2 and 3 only consider a snapshot representation of the system, disregarding whether there has been a temporal evolution of the network; on the other, the work in Chapter 4 observes the graph at different instants, describing the evolution *in time* of the system.

In conclusion, while dealing with the analysis of social networks, this

work provides an overview of different but complementary methodologies to describe these systems. The main differences reside in the “philosophy” behind the model construction, i.e. whether the topology or other nodes-related features may provide a more accurate benchmark, and in the temporal representation of the graph. We have provided theoretical tools and real-world applications of both cases and the obtained results show that both these approaches turn out to be valuable in reproducing several aspects of the systems under analysis.

Appendix A

This Chapter provides the details regarding the development of each null model used in Chapter 2. First of all we explain the passages of the Bipartite Score CM computation. Then we proceed explaining the calculations for all the proposed alternative models, i.e. the Weighted CM with truncated sequence of possible weights, the Partial Score CM and Random Graph CM. Finally, we show the results of some other performed analyses that have not been included in the main body of the work.

A.1 Bipartite Score Configuration Model

The general formulation of the probability distribution over the ensemble of graphs is given by

$$P(\mathbf{G}|\vec{\eta}_\beta, \vec{\theta}_\beta) = \frac{\exp(-H(\mathbf{G}, \vec{\eta}_\beta, \vec{\theta}_\beta))}{Z(\vec{\eta}_\beta, \vec{\theta}_\beta)} \quad (\text{A.1})$$

where the term $H(\mathbf{G})$ is the *Hamiltonian* of the problem and Z is a normalizing factor called *partition function*. Imposing the constraints on the

degree sequence in the Hamiltonian formulation we get

$$\begin{aligned}
H(\mathbf{G}, \vec{\eta}_\beta, \vec{\theta}_\beta) &= \sum_\beta \left(\sum_i k_{i,\beta}(\mathbf{G}) \cdot \eta_{i,\beta} + \sum_\alpha k_{\alpha,\beta}(\mathbf{G}) \cdot \theta_{\alpha,\beta} \right) \\
&= \sum_\beta \left(\sum_{i,\alpha} m_{i,\alpha,\beta} \eta_{i,\beta} + \sum_{i,\alpha} m_{i,\alpha,\beta} \theta_{\alpha,\beta} \right) \\
&= \sum_\beta \sum_{i,\alpha} m_{i,\alpha,\beta} (\eta_{i,\beta} + \theta_{\alpha,\beta}),
\end{aligned} \tag{A.2}$$

meaning that we impose the degree values per ratings have to be preserved and be equal to their expected values in the ensemble. The computation of the partition function returns instead

$$\begin{aligned}
Z(\vec{\eta}_\beta, \vec{\theta}_\beta) &= \sum_{\mathbf{G} \in \mathcal{G}} \exp \left(- \sum_\beta \sum_{i,\alpha} (\eta_{i,\beta} + \theta_{\alpha,\beta}) m_{i,\alpha,\beta} \right) \\
&= \prod_{i,\alpha} \sum_{\mathbf{G} \in \mathcal{G}} \exp \left(- \sum_\beta (\eta_{i,\beta} + \theta_{\alpha,\beta}) m_{i,\alpha,\beta} \right) \\
&= \prod_{i,\alpha} \left[1 + \sum_\beta \exp(-\eta_{i,\beta} - \theta_{\alpha,\beta}) \right]
\end{aligned} \tag{A.3}$$

where the last step is justified by the fact that all $m_{i,\alpha,\beta}$ are mutually exclusive, so the presence of an edge with rating $\hat{\beta}$ excludes all the others (i.e. $m_{i,\alpha,\hat{\beta}} = 1$ implies $m_{i,\alpha,\beta} = 0$ for all $\beta \neq \hat{\beta}$). Implementing equations (A.2) and (A.3) in (A.1) and using the substitution $x_{i,\beta} = e^{-\eta_{i,\beta}}$ and $y_{\alpha,\beta} = e^{-\theta_{\alpha,\beta}}$ we get (2.5), as follows

$$\begin{aligned}
P(\mathbf{G}|\vec{x}, \vec{y}) &= \prod_{i,\alpha} \frac{\exp \left(- \sum_\beta m_{i,\alpha,\beta} (\eta_{i,\beta} + \theta_{\alpha,\beta}) \right)}{1 + \sum_\beta \exp(-\eta_{i,\beta} - \theta_{\alpha,\beta})} \\
&= \prod_{i,\alpha} \frac{\prod_\beta (x_{i,\beta} y_{\alpha,\beta})^{m_{i,\alpha,\beta}}}{1 + \sum_\beta x_{i,\beta} y_{\alpha,\beta}}
\end{aligned} \tag{A.4}$$

Finally, the log-likelihood in equation (2.8) can be obtained simply computing $\mathcal{L}(\vec{x}, \vec{y}|\mathbf{G}^*) = \log P(\mathbf{G}^*|\vec{x}, \vec{y})$ while the system is obtained as the

first order conditions of equation (2.8), as follows

$$\begin{aligned}\frac{\partial \mathcal{L}(\vec{x}, \vec{y} | \mathbf{G}^*)}{\partial x_{i,\beta}} &= \frac{k_{i,\beta}(\mathbf{G}^*)}{x_{i,\beta}} - \sum_{\alpha} \frac{y_{\alpha,\beta}}{1 + \sum_{\beta} x_{i,\beta} y_{\alpha,\beta}} = 0 \quad \text{for all } i, \beta \\ \frac{\partial \mathcal{L}(\vec{x}, \vec{y} | \mathbf{G}^*)}{\partial y_{\alpha,\beta}} &= \frac{k_{\alpha,\beta}(\mathbf{G}^*)}{y_{\alpha,\beta}} - \sum_i \frac{x_{i,\beta}}{1 + \sum_{\beta} x_{i,\beta} y_{\alpha,\beta}} = 0 \quad \text{for all } \alpha, \beta\end{aligned}\tag{A.5}$$

Rearranging the terms in the previous equation (A.5) we immediately obtain the analytical expression for the expected degree of each node, that corresponds to the system in equation (2.9).

A.2 Bipartite Weighted Configuration Model with truncated sequence of weights

We here propose a new formulation for the Weighted CM. We denote it with the term *truncated* since, in the presented framework, the possible weights assume integer values and only a finite range of weights are allowed. Therefore, the CM formulation has to be modified accordingly.

The general formulation of the probability distribution over the ensemble of graphs is given by

$$P(\mathbf{G} | \vec{\eta}, \vec{\theta}) = \frac{\exp(-H(\mathbf{G}, \vec{\eta}, \vec{\theta}))}{Z(\vec{\eta}, \vec{\theta})}\tag{A.6}$$

In this case, we need to impose the observed strength sequence as a constraint. By doing so, the expected strength of each node in the ensemble is preserved equal to its observed value. Following the notation introduced in Chapter 2, the strengths are defined as follows

$$\begin{aligned}s_i(\mathbf{G}) &= \sum_{\alpha} m_{i,\alpha} \quad \text{for all } i \in L \\ s_{\alpha}(\mathbf{G}) &= \sum_i m_{i,\alpha} \quad \text{for all } \alpha \in \Gamma\end{aligned}\tag{A.7}$$

Therefore, the Hamiltonian of the problem reads as

$$\begin{aligned}
H(\mathbf{G}, \vec{\eta}, \vec{\theta}) &= \sum_i s_i(\mathbf{G}) \cdot \eta_i + \sum_{\alpha} s_{\alpha}(\mathbf{G}) \cdot \theta_{\alpha} \\
&= \sum_{i,\alpha} m_{i,\alpha} \eta_i + \sum_{i,\alpha} m_{i,\alpha} \theta_{\alpha} \\
&= \sum_{i,\alpha} m_{i,\alpha} (\eta_i + \theta_{\alpha}).
\end{aligned} \tag{A.8}$$

Using the substitution $x_i = e^{-\eta_i}$ and $y_{\alpha} = e^{-\theta_{\alpha}}$, the partition function becomes

$$\begin{aligned}
Z(\vec{\eta}, \vec{\theta}) &= \sum_{\mathbf{G} \in \mathcal{G}} \exp \left(- \sum_{i,\alpha} m_{i,\alpha} (\eta_i + \theta_{\alpha}) \right) \\
&= \prod_{i,\alpha} \sum_{\mathbf{G} \in \mathcal{G}} \exp \left(m_{i,\alpha} (\eta_i + \theta_{\alpha}) \right) \\
&= \prod_{i,\alpha} \sum_{m_{i,\alpha}} (x_i y_{\alpha})^{m_{i,\alpha}} \\
&= \prod_{i,\alpha} \frac{1 - (x_i y_{\alpha})^{\beta_{max}+1}}{1 - x_i y_{\alpha}}.
\end{aligned} \tag{A.9}$$

It is worth to notice that the last two passages hold since the scores are mutually exclusive, therefore $m_{i,\alpha,\hat{\beta}} = 1$ implies $m_{i,\alpha,\beta} = 0$ for all $\beta \neq \hat{\beta}$. Implementing equations (A.8) and (A.9) into (A.6) we obtain the following probability distribution over the graphs ensemble

$$P(\mathbf{G}|\vec{x}, \vec{y}) = \prod_{i,\alpha} \frac{(1 - x_i y_{\alpha})(x_i y_{\alpha})^{m_{i,\alpha}}}{1 - (x_i y_{\alpha})^{\beta_{max}+1}}. \tag{A.10}$$

where the terms

$$p_{i,\alpha,\beta} = \frac{(1 - x_i y_{\alpha})(x_i y_{\alpha})^{\beta}}{1 - (x_i y_{\alpha})^{\beta_{max}+1}}. \tag{A.11}$$

represent the probability to observe a link with rating β between nodes i and α . Notice that the previous equation (A.11) identifies a truncated geometric distribution with parameter $x_i y_{\alpha}$ and $\beta \in \{1, \dots, \beta_{max}\}$. The log-likelihood of the problem can be obtained as $\mathcal{L}(\vec{x}, \vec{y}|\mathbf{G}^*) = \log P(\mathbf{G}^*|\vec{x}, \vec{y})$

and takes the following form

$$\begin{aligned}\mathcal{L}(\vec{x}, \vec{y} | \mathbf{G}^*) &= \sum_i s_i(\mathbf{G}^*) \log(x_i) + \sum_\alpha s_\alpha(\mathbf{G}^*) \log(y_\alpha) \\ &\quad + \sum_{i,\alpha} \log(1 - x_i y_\alpha) - \sum_{i,\alpha} \log [1 - (x_i y_\alpha)^{\beta_{max}+1}].\end{aligned}\tag{A.12}$$

Taking the partial derivatives of the log-likelihood we get

$$\begin{aligned}\frac{\partial \mathcal{L}(\vec{x}, \vec{y} | \mathbf{G}^*)}{\partial x_i} &= \frac{s_i(\mathbf{G}^*)}{x_i} - \sum_\alpha \frac{y_\alpha}{1 - x_i y_\alpha} + \sum_\alpha \frac{(\beta_{max} + 1) y_\alpha (x_i y_\alpha)^{\beta_{max}}}{1 - (x_i y_\alpha)^{\beta_{max}+1}} \\ \frac{\partial \mathcal{L}(\vec{x}, \vec{y} | \mathbf{G}^*)}{\partial y_\alpha} &= \frac{s_\alpha(\mathbf{G}^*)}{y_\alpha} - \sum_i \frac{x_i}{1 - x_i y_\alpha} + \sum_i \frac{(\beta_{max} + 1) x_i (x_i y_\alpha)^{\beta_{max}}}{1 - (x_i y_\alpha)^{\beta_{max}+1}}\end{aligned}\tag{A.13}$$

for all $i \in L$ and $\alpha \in \Gamma$. Setting the previous equations equal to zero and rearranging the terms we easily obtain an analytical expression for the expected strength sequence

$$\begin{aligned}s_i(\mathbf{G}^*) &= \sum_\alpha \frac{x_i y_\alpha [\beta_{max}(x_i y_\alpha)^{\beta_{max}+1} - (\beta_{max} + 1)(x_i y_\alpha)^{\beta_{max}} + 1]}{(1 - x_i y_\alpha) [1 - (x_i y_\alpha)^{\beta_{max}+1}]} \\ s_\alpha(\mathbf{G}^*) &= \sum_i \frac{x_i y_\alpha [\beta_{max}(x_i y_\alpha)^{\beta_{max}+1} - (\beta_{max} + 1)(x_i y_\alpha)^{\beta_{max}} + 1]}{(1 - x_i y_\alpha) [1 - (x_i y_\alpha)^{\beta_{max}+1}]}\end{aligned}\tag{A.14}$$

for all $i \in L$ and $\alpha \in \Gamma$.

A.3 Bipartite Partial Score Configuration Model

In this section we provide a less-constrained version of the Bipartite Score CM, called for this reason *Partial Score CM*. Instead of constraining on the entire observed degree sequence, we only impose as constraints the empirical degree sequence observed in one of the layers, for each rating level.

The probability distribution over the ensemble of graphs is given by

$$P(\mathbf{G}|\vec{\eta}_\beta) = \frac{\exp(-H(\mathbf{G}, \vec{\eta}_\beta))}{Z(\vec{\eta}_\beta)} \quad (\text{A.15})$$

and the expression for the Hamiltonian reads as follows

$$H(\mathbf{G}, \vec{\eta}_\beta) = \sum_{i,\beta} k_{i,\beta}(\mathbf{G}) \cdot \eta_{i,\beta} = \sum_{\beta} \sum_{i,\alpha} m_{i,\alpha,\beta} \eta_{i,\beta} \quad (\text{A.16})$$

Notice that in the previous equation (A.16) we include the sequence $\{k_{i,\beta}\}_{i \in L}$ for all β as constraints of the problem, that represents the degree sequence for nodes in L for each score level. The Hamiltonian is equivalently defined with the degree sequence on the other layer Γ . The partition function of the problem is instead

$$\begin{aligned} Z(\vec{\eta}_\beta) &= \sum_{\mathbf{G} \in \mathcal{G}} \exp \left(- \sum_{\beta} \sum_{i,\alpha} m_{i,\alpha,\beta} \eta_{i,\beta} \right) \\ &= \prod_{i,\alpha} \sum_{\mathbf{G} \in \mathcal{G}} \exp \left(- \sum_{\beta} m_{i,\alpha,\beta} \eta_{i,\beta} \right) \\ &= \prod_{i,\alpha} \left[1 + \sum_{\beta} \exp(-\eta_{i,\beta}) \right] \end{aligned} \quad (\text{A.17})$$

Again, the last passage is justified by the fact the all betas are mutually exclusive. Plugging equations (A.16) and (A.17) into (A.15) and using the substitution $e^{-\eta_{i,\beta}} = x_{i,\beta}$ we obtain the following probability distribution over the graphs ensemble

$$\begin{aligned} P(\mathbf{G}|\vec{x}) &= \prod_{i,\alpha} \frac{\exp \left(- \sum_{\beta} m_{i,\alpha,\beta} \eta_{i,\beta} \right)}{1 + \sum_{\beta} \exp(-\eta_{i,\beta})} \\ &= \prod_{i,\alpha} \frac{\prod_{\beta} (x_{i,\beta})^{m_{i,\alpha,\beta}}}{1 + \sum_{\beta} x_{i,\beta}} \\ &= \prod_i \frac{\prod_{\beta} (x_{i,\beta})^{k_{i,\beta}}}{\left(1 + \sum_{\beta} x_{i,\beta} \right)^{N_\Gamma}} \end{aligned} \quad (\text{A.18})$$

where the single terms

$$p_{i,\beta} = \frac{x_{i,\beta}}{1 + \sum_{\beta} x_{i,\beta}} = \frac{k_{i,\beta}}{N_\Gamma} \quad (\text{A.19})$$

simply identifies the probability to observe a link with score β incident to node i and coincides with the empirical frequency observed for score β . Notice that the previous probability in equation (A.19) is invariant for $\alpha \in \Gamma$. Therefore, each edge (i, α) with score β can be observed with constant probability for all $\alpha \in \Gamma$. As in the previous cases, the log-likelihood of the problem can be obtained as $\mathcal{L}(\vec{x}|\mathbf{G}^*) = \log P(\mathbf{G}^*|\vec{x})$ and takes the following expression

$$\mathcal{L}(\vec{x}|\mathbf{G}^*) = \sum_{i,\beta} k_{i,\beta}(\mathbf{G}^*) \log(x_{i,\beta}) - \sum_i N_\Gamma \log\left(1 + \sum_\beta x_{i,\beta}\right) \quad (\text{A.20})$$

Finally, the First Order Conditions of equation (A.20) are

$$\frac{\partial \mathcal{L}(\vec{x}|\mathbf{G}^*)}{\partial x_{i,\beta}} = \frac{k_{i,\beta}(\mathbf{G}^*)}{x_{i,\beta}} - \frac{N_\Gamma}{1 + \sum_\beta x_{i,\beta}} = 0 \text{ for all } i, \beta \quad (\text{A.21})$$

and rearranging the terms we immediately obtain the analytical expression for the expected degrees, as

$$k_{i,\beta}(\mathbf{G}^*) = \frac{N_\Gamma x_{i,\beta}}{1 + \sum_\beta x_{i,\beta}} = \langle k_{i,\beta} \rangle \text{ for all } i, \beta. \quad (\text{A.22})$$

In other words, the expected number of links with score β incident to node i can be obtained multiplying the number of nodes on layer Γ by the probability to observe an edge with score β .

A.4 Bipartite Score Random Graph

In the last section we propose a generalisation of the Random Graph CM adapted to the case of rating networks. In this case, the entropy is maximised under the set of constraints specifying the total number of edges observed for each score level, denoted as E_β .

Again, the probability distribution over the ensemble is given by

$$P(\mathbf{G}|\vec{\theta}) = \frac{\exp\left(-H(\mathbf{G}, \vec{\theta})\right)}{Z(\vec{\theta})} \quad (\text{A.23})$$

where the Hamiltonian of the problem is given by

$$H(\mathbf{G}, \vec{\theta}) = \sum_{\beta} \theta_{\beta} \cdot E_{\beta}(\mathbf{G}) = \sum_{\beta} \theta_{\beta} \sum_{i, \alpha} m_{i, \alpha, \beta} \quad (\text{A.24})$$

Notice that the observed number of edges for each rating level is preserved in the ensemble, i.e. is fixed equal to its expected value. The partition function takes the form

$$\begin{aligned} Z(\vec{\theta}) &= \sum_{\mathbf{G} \in \mathcal{G}} \exp \left(- \sum_{\beta} \theta_{\beta} \sum_{i, \alpha} m_{i, \alpha, \beta} \right) \\ &= \prod_{i, \alpha} \sum_{\mathbf{G} \in \mathcal{G}} \exp \left(- \sum_{\beta} \theta_{\beta} m_{i, \alpha, \beta} \right) \\ &= \prod_{i, \alpha} \left[1 + \sum_{\beta} \exp(-\theta_{\beta}) \right] \end{aligned} \quad (\text{A.25})$$

Again, the last two passages hold since the score levels are mutually exclusive. Plugging equations (A.24) and (A.25) into (A.23) and using the substitution $e^{-\theta_{\beta}} = x_{\beta}$ we obtain the following expression for the probability distribution over the ensemble of graphs

$$\begin{aligned} P(\mathbf{G} | \vec{x}) &= \prod_{i, \alpha} \frac{\exp \left(- \sum_{\beta} \theta_{\beta} m_{i, \alpha, \beta} \right)}{1 + \sum_{\beta} \exp(-\theta_{\beta})} \\ &= \prod_{i, \alpha} \frac{\prod_{\beta} (x_{\beta})^{m_{i, \alpha, \beta}}}{1 + \sum_{\beta} x_{\beta}} \\ &= \frac{\prod_{\beta} (x_{\beta})^{E_{\beta}}}{\left(1 + \sum_{\beta} x_{\beta} \right)^{N_L N_{\Gamma}}} \end{aligned} \quad (\text{A.26})$$

where the terms

$$p_{\beta} = \frac{x_{\beta}}{1 + \sum_{\beta} x_{\beta}} = \frac{E_{\beta}}{N_L N_{\Gamma}} \quad (\text{A.27})$$

denotes the probability to observe a link of score β between any pair of nodes. Notice that the previous probability is invariant for all pairs of nodes and coincides with the empirical frequency of observed edges for the considered rating. As in the previous cases, the log-likelihood

associated to equation (A.26) can be computed as $\mathcal{L}(\vec{x}|\mathbf{G}^*) = \log P(\mathbf{G}^*|\vec{x})$ and in this case takes the following form

$$\mathcal{L}(\vec{x}|\mathbf{G}^*) = \sum_{\beta} E_{\beta}(\mathbf{G}^*) \log(x_{\beta}) - N_L N_{\Gamma} \log\left(1 + \sum_{\beta} x_{\beta}\right) \quad (\text{A.28})$$

The First Order Conditions associated to equation (A.28) can be computed as

$$\frac{\partial \mathcal{L}(\vec{x}|\mathbf{G}^*)}{\partial x_{\beta}} = \frac{E_{\beta}(\mathbf{G}^*)}{x_{\beta}} - \frac{N_L N_{\Gamma}}{1 + \sum_{\beta} x_{\beta}} = 0 \text{ for all } \beta \quad (\text{A.29})$$

and rearranging the terms we can easily obtain the analytical expression for the expected number of edges

$$E_{\beta}(\mathbf{G}^*) = \frac{N_L N_{\Gamma} x_{\beta}}{1 + \sum_{\beta} x_{\beta}} = \langle E_{\beta} \rangle \text{ for all } \beta \quad (\text{A.30})$$

In other words, the expected number of edges with rating β can be obtained simply multiplying the total number of possible links $N_L N_{\Gamma}$ by the probability to observe a link with score β .

A.5 Further analyses of higher topological quantities: positive reviews only

In order to check the capability of the model of reproducing just positive ratings, we use a threshold score β_{th} to construct a binarized version of the original matrix, $\mathbf{M}^o = \{m_{i,\alpha}^o : i \in L, \alpha \in \Gamma\}$. In this case the matrix entries can assume values $m_{i,\alpha}^o \in \{1, 0\}$. However, while the entry $m_{i,\alpha}^o = 1$ indicates the presence of a positive (i.e. $\beta_{th} \leq \beta \leq \beta_{max}$) review in the original data, the other value $m_{i,\alpha}^o = 0$ is used for both the absence of a link and the presence of a negative review (i.e. $1 \leq \beta < \beta_{th}$). The associated probability matrix is indicated as $\langle \mathbf{M}^o \rangle$, which entries $\langle m_{i,\alpha}^o \rangle$ represent the probability to observe a score greater than the threshold from user i to node α . In this setting, we use the standard definition of node degrees,

$$k_i(\mathbf{M}^o) = \sum_{\alpha} m_{i,\alpha}^o \text{ for all } i \in L$$

and the correlation between neighbors' degrees is studied with the classical definition of average-nearest-neighbour degree,

$$k_i^{nn}(\mathbf{M}^o) = \frac{\sum_{\alpha} \sum_j m_{i,\alpha}^o m_{j,\alpha}^o}{\sum_{\alpha} m_{i,\alpha}^o} \text{ for all } i \in L.$$

The results are presented in Figures 35-37. For these applications we have considered $\beta_{th} = 3$. In all cases our method is perfectly able to identify the data's overall trend. Despite few observations still lie outside the confidence region, we can state that the correlation between neighbors' degrees may be interpreted as a consequence of the network topology, whenever the full distribution of received scores is specified in the null model construction. On the other hand, the other ensembles significantly underestimate the observed traces (PCM and RG for all the datasets, WCM only for the Amazons'), leading to the conclusion that their underlying constraints are not significantly explanatory to reproduce the present assortativity patterns.

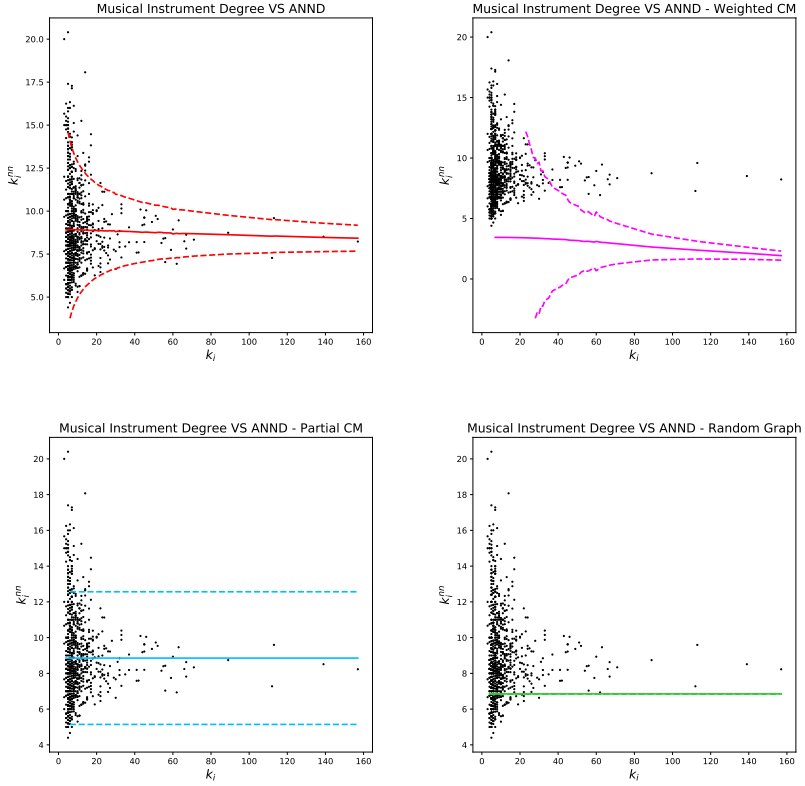


Figure 35: Application of the method to the MI network. All plots represent k_i^{nn} versus k_i . Red lines show the expectation values computed with our method. Magenta, blue and green lines are instead the expectation values under WCM, PCM and Erdős-Rényi RG. The area of ± 2 standard deviations around the average value is represented in dashed linestyle.

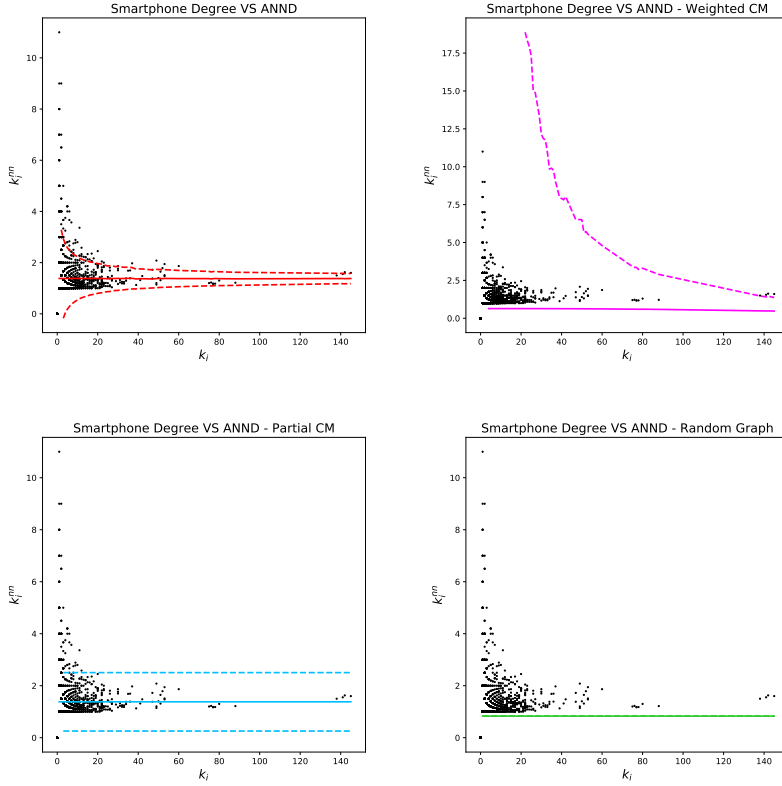


Figure 36: Application of the method to the SM network. All plots represent k_i^{nn} versus k_i . Red lines show the expectation values computed with our method. Magenta, blue and green lines are instead the expectation values under WCM, PCM and Erdős-Rényi RG. The area of ± 2 standard deviations around the average value is represented in dashed linestyle.

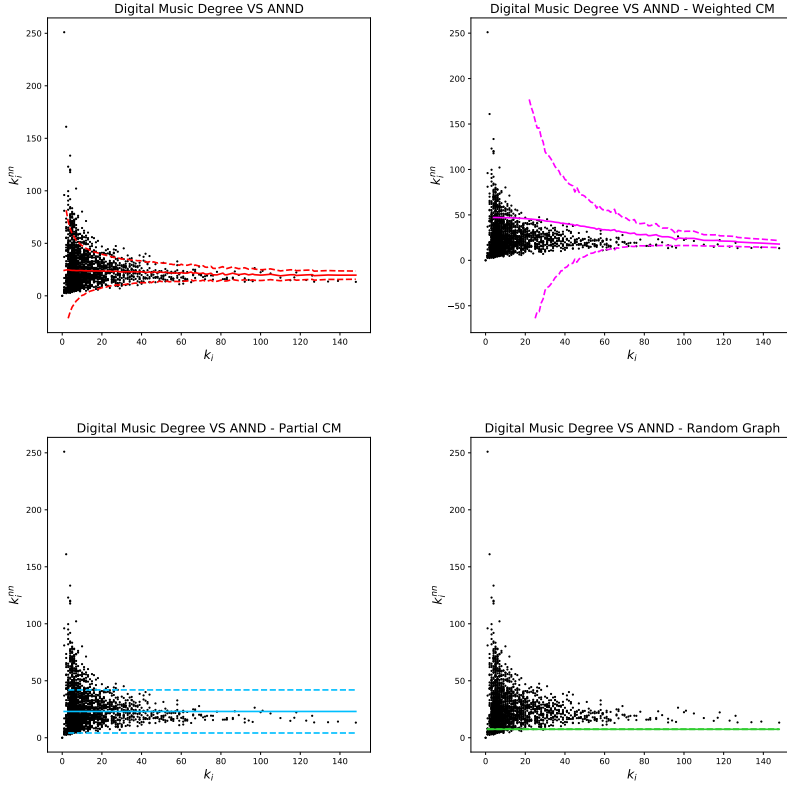


Figure 37: Application of the method to the DM network. All plots represent k_i^{nm} versus k_i . Red lines show the expectation values computed with our method. Magenta, blue and green lines are instead the expectation values under WCM, PCM and Erdős-Rényi RG. The area of ± 2 standard deviations around the average value is represented in dashed linestyle.

Appendix B

This Chapter collects additional information that were not included into the main body of Chapter 3, in order to ease the presentation of the results. First we review the steps employed to construct an entropy-based null model for a bipartite and directed network, as presented in [104]. Then we provide additional figures and tables, with relative captions, not included in the main body of the Chapter.

B.1 Bipartite Directed Configuration Model

As mentioned in Chapter 3, we represent the tweeting and retweeting activity by means of a bipartite and directed network $\mathbf{G}_{\text{BiD}}^* = (U, P, \mathcal{T}, \mathcal{R})$, that collects users and tweets in the two layers. Using the same notation as in Section 3.3.3, an outgoing edge $(i, p) \in \mathcal{T}$ indicates that user i has tweeted post p on the online social network, while an edge in the opposite direction $(p, j) \in \mathcal{R}$ shows that $j \neq i$ has retweeted the same post at least once during the election days.

The authors in [104] show that the structure of one bipartite and directed network can be entirely described with *two* biadjacency matrices, one for the tweets, i.e. the matrix \mathbf{T}^* , and another for the retweets, i.e. the matrix \mathbf{R}^* , that are the two off-diagonal blocks of the matrix in equation (3.4). At this point, the two bipartite simple networks can be separately randomized by means of the standard Bipartite Configuration Mo-

del [36]. We sketch the passages here in order to introduce the notation. Let

$$\begin{aligned} H(\mathbf{T}, \vec{\eta}, \vec{\theta}) &= \sum_{i \in U} \eta_i k_i^{out} + \sum_{p \in P} \theta_p k_p^{in} \\ &= \sum_{i,p} (\eta_i + \theta_p) t_{i,p} \end{aligned}$$

be the Hamiltonian of the problem for the biadjacency matrix of the tweets \mathbf{T}^* , in which we have imposed a set of constraints on the outgoing degrees of the users and the in-going degrees of the tweets. Essentially, those two quantities represent the number of tweets posted by a single user and the number of authors of a tweet. In the main text we have described how this second quantity is trivially equal to one for all posts and the consequences of this simplification. However, herefrom we keep the notation general. The parameters of the problem $(\vec{\eta}, \vec{\theta})$ have respectively dimension N_U and N_P .

A similar expression can be obtained for the biadjacency matrix of the retweets \mathbf{R}^* , but in this case the two alternative quantities k_i^{in} and k_p^{out} need to be fixed, representing respectively the number of retweets performed by a user and received by a tweet. In this case the Hamiltonian of the problem is the following

$$\begin{aligned} H(\mathbf{R}, \vec{\eta}', \vec{\theta}') &= \sum_{i' \in U} \eta'_{i'} k_{i'}^{in} + \sum_{p' \in P} \theta'_{p'} k_{p'}^{out} \\ &= \sum_{i', p'} (\eta'_{i'} + \theta'_{p'}) r_{p', i'} \end{aligned}$$

The partition function for the matrix \mathbf{T}^* is given by

$$\begin{aligned} Z(\vec{\eta}, \vec{\theta}) &= \sum_{\mathbf{T}} \exp \left[- \sum_{i,p} (\eta_i + \theta_p) t_{i,p} \right] \\ &= \prod_{i,p} \sum_{\mathbf{T}} \exp [- (\eta_i + \theta_p) t_{i,p}] \\ &= \prod_{i,p} [1 + \exp(\eta_i + \theta_p)] \end{aligned}$$

Therefore, the probability distribution over the ensemble of bipartite graphs can be expressed as

$$\begin{aligned}
P(\mathbf{T}|\vec{\eta}, \vec{\theta}) &= \prod_{i,p} \frac{\exp [-(\eta_i + \theta_p)t_{i,p}]}{1 + \exp(-\eta_i - \theta_p)} \\
&= \prod_{i,p} \frac{(z_i \zeta_p)^{t_{i,p}}}{1 + z_i \zeta_p} \\
&= \prod_{i,p} q_{i,p}^{t_{i,p}} (1 - q_{i,p})^{1-t_{i,p}}
\end{aligned} \tag{B.1}$$

where the terms

$$q_{i,p} = \frac{z_i \zeta_p}{1 + z_i \zeta_p}$$

represent the probability to observe the directed link (i, p) and are obtained after the substitution $\exp(-\eta_i) = z_i$ and $\exp(-\theta_p) = \zeta_p$.

Analogously, starting from the expression of the Hamiltonian for the bipartite graph \mathbf{R}^* , the following partition function can be recovered

$$\begin{aligned}
Z(\vec{\eta}', \vec{\theta}') &= \sum_{\mathbf{R}} \exp \left[- \sum_{i',p'} (\eta'_{i'} + \theta'_{p'}) r_{p',i'} \right] \\
&= \prod_{i',p'} \sum_{\mathbf{R}} \exp [-(\eta'_{i'} + \theta'_{p'}) r_{p',i'}] \\
&= \prod_{i',p'} [1 + \exp(\eta'_{i'} + \theta'_{p'})]
\end{aligned}$$

while the probability distribution over this second ensemble of bipartite graphs can be expressed as

$$\begin{aligned}
P(\mathbf{R}|\vec{\eta}', \vec{\theta}') &= \prod_{i',p'} \frac{\exp [-(\eta'_{i'} + \theta'_{p'}) r_{p',i'}]}{1 + \exp(-\eta'_{i'} - \theta'_{p'})} \\
&= \prod_{i',p'} \frac{(z'_{i'} \zeta'_{p'})^{r_{p',i'}}}{1 + z'_{i'} \zeta'_{p'}} \\
&= \prod_{i',p'} (q'_{p',i'})^{r_{p',i'}} (1 - q'_{p',i'})^{1-r_{p',i'}}
\end{aligned} \tag{B.2}$$

where, as in the previous case, the terms

$$q'_{p',i'} = \frac{z'_{i'} \zeta'_{p'}}{1 + z'_{i'} \zeta'_{p'}}$$

represent the probability to observe the directed link (p', i') and are obtained after the substitution $\exp(-\eta'_{i'}) = z'_{i'}$ and $\exp(-\theta'_{p'}) = \zeta'_{p'}$.

Therefore, as stated in Chapter 3, the product of the two terms in equations (B.1) and (B.2) leads to the full expression of the probability distribution defined over the ensemble of bipartite and directed graphs, provided in equation (3.5). It is worth noticing that the two vectors of multipliers \vec{z} and \vec{z}' (and equivalently for $\vec{\zeta}$ and $\vec{\zeta}'$) are associated to the same set of nodes in the two models, i.e. the set of users (tweets). However, the different notation is due to the fact that different constraints are imposed in the two frameworks.

In order to determine a numerical value for the involved Lagrangian multipliers, the following log-likelihood functions need to be maximized

$$\begin{aligned} \mathcal{L}(\vec{z}, \vec{\zeta} | \mathbf{T}^*) &= \sum_i k_i^{out}(\mathbf{T}^*) \ln(z_i) + \sum_p k_p^{in} \ln(\zeta_p) - \sum_{i,p} \ln(1 + z_i \zeta_p) \\ \mathcal{L}(\vec{z}', \vec{\zeta}' | \mathbf{R}^*) &= \sum_{i'} k_{i'}^{in}(\mathbf{R}^*) \ln(z'_{i'}) + \sum_{p'} k_{p'}^{out}(\mathbf{R}^*) \ln(\zeta'_{p'}) - \sum_{i',p'} \ln(1 + z'_{i'} \zeta'_{p'}) \end{aligned} \quad (\text{B.3})$$

or, equivalently, one needs to choose the values $(\vec{z}^*, \vec{\zeta}^*)$ and $((\vec{z}')^*, (\vec{\zeta}')^*)$ that solve the following systems of coupled equations

$$\begin{aligned} \langle k_i^{out} \rangle &= \sum_{p \in P} q_{i,p} = k_i^{out}(\mathbf{T}^*) \quad \forall i \in U \\ \langle k_p^{in} \rangle &= \sum_{i \in U} q_{i,p} = k_p^{in}(\mathbf{T}^*) \quad \forall p \in P \end{aligned} \quad (\text{B.4})$$

and

$$\begin{aligned} \langle k_{i'}^{in} \rangle &= \sum_{p' \in P} q'_{p',i'} = k_{i'}^{in}(\mathbf{R}^*) \quad \forall i' \in U \\ \langle k_{p'}^{out} \rangle &= \sum_{i' \in U} q'_{p',i'} = k_{p'}^{out}(\mathbf{R}^*) \quad \forall p' \in P \end{aligned} \quad (\text{B.5})$$

B.2 Additional Tables and Figures

Figure 38 represents the distribution of the polarization values observed for users with the same number of interactions. Also in this case, the distribution is skewed towards the higher values, indicating a focus towards the same group of users.

Table 15 represents a list of the most central users in the validated network of verified users.

Tables 16 and 17 provide a list of the most retweeted users in the projected network of retweets, while Tables 18 and 19 list the most active users in terms of retweeting posts.

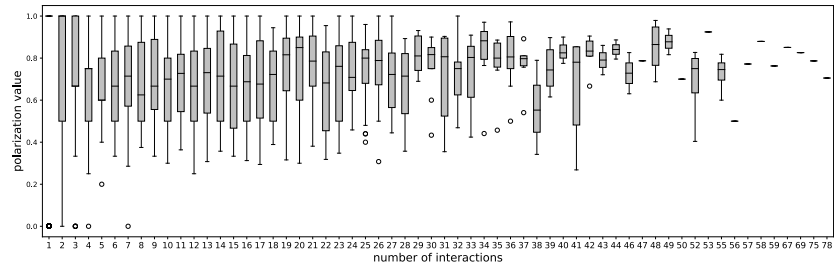


Figure 38: Distribution of the polarization indices, conditioned on the total number of interactions observed for each users.

community	users	centrality index
●	la Repubblica	0.0532
●	Agenzia ANSA	0.0382
●	Il Fatto Quotidiano	0.0300
●	Sky TG24	0.0292
●	Corriere della Sera	0.0285
●	YouTrend	0.0247
●	Rainews	0.0228
●	Peter Gomez	0.0152
●	Il Post	0.0142
●	IlSole24ORE	0.0137
●	Matteo Renzi	0.0133
●	Tg La7	0.0124
●	Ferruccio de Bortoli	0.0117
●	Marco Castelnuovo	0.0104
●	Internazionale	0.0100
●	Il Foglio	0.0100
●	Lorenzo Pregliasco	0.0095
●	Confindustria	0.0093
●	Agi Agenzia Italia	0.0086
●	Vittorio Zucconi	0.0085
●	Selvaggia Lucarelli	0.0072
●	Pietro Grasso	0.0071
●	Tgcom24	0.0069
●	Partito Democratico	0.0067
●	diMartedì	0.0066
●	La Stampa	0.0066
●	Stefano Feltri	0.0066
●	L'Espresso	0.0065
●	Giuseppe Smorto	0.0063
●	Ficarra e Picone	0.0062
●	Rai Radio1	0.0061
●	Il Viminale	0.0061
●	Ezio Mauro	0.0057
●	Movimento 5 Stelle	0.0057
●	Anna Ascani	0.0055
●	insopportabile	0.0055
●	Agenzia DIRE	0.0054
●	Matteo Salvini	0.0053
●	Fabio Chiusi	0.0052
●	Riccardo Puglisi	0.0052
●	Luca Tentoni	0.0051
●	Paolo Attivissimo	0.0045
●	RTL 102.5	0.0045
●	Franz Russo	0.0044
●	Matteo Pedrosi	0.0042
●	Marco Travaglio	0.0041
●	trashitaliano.it	0.0041
●	Tommaso Labate	0.0040
●	Giorgio La Porta	0.0040
●	Sebastiano Messina	0.0038

Table 15: List of the most central users in the validated network of verified users. The coloured dot left to the name indicates the political affiliation of the user according to our division in community.


Community 	Community 
Il Fatto Quotidiano ✓	la Repubblica ✓
Peter Gomez ✓	Agenzia ANSA ✓
Non Voto il PD	trashitaliano.it ✓
Elio Lannutti	Sky TG24 ✓
Manlio Di Stefano ✓	YouTrend ✓
Kid Stardust	Corriere della Sera ✓
Marco Travaglio ✓	Il Post ✓
Antonio Bordin	Rainews ✓
Ficarra e Picone ✓	Tg La7 ✓
Movimento 5 Stelle ✓	Pietro Grasso ✓
dukana	Rolling Stone Italia ✓
Damiano	Repubblica Tv
Sere Vuole Votare	sheldøn
luisella costamagna	Paola.
Arsenale K	Fran Altomare ✓
Selvaggia Lucarelli ✓	Homme à projets
Emilio Carelli	Tg2
Ferruccio de Bortoli ✓	Rai Radio2 ✓
sicut vuole votare	Alessandro Masala ✓
carlo sibilìa ✓	Giuseppe Civati ✓
Luca Miniero	Donna Chisciotte
Commentatore Libero	Alessandro Alciato ✓
Nicola Morra ✓	Daniela Collu ✓

Table 16: Following the approach of [78] we build the monopartite and directed network of retweets, in which an edge (i, j) indicates that j has retweeted i at least once. Then we select the directed subgraph generated by each community and here we report the list of the most retweeted users within the subgraphs of M5S and information channels communities. The symbol ✓ next to the name indicates that the account has been verified.

Community ●	Community ●
Matteo Renzi ✓	Matteo Salvini ✓
Partito Democratico ✓	Claudio Borghi A. ✓
Vittorio Zucconi ✓	Giorgio La Porta ✓
Claudio Cerasa	Giorgia Meloni ✓
Eugenio Cardi	Silvio Berlusconi ✓
IlSole24ORE ✓	Dr. Alice Weidel ✓
Sebastiano Messina ✓	Patrizia Rametta
Alessia Morani ✓	Forza Italia ✓
Il Foglio ✓	Lega - Salvini Premier ✓
Pietro Raffa ✓	CasaPound Italia
jacopo iacoboni	Barbara Raval
Riccardo Magi ✓	Simone Di Stefano
Daniele Cinà	Renato Brunetta ✓
mattia feltri ✓	Ale
Lorenzo Pregliasco ✓	Alessandro Meluzzi
Anna Ascani ✓	SONIA GROTTA
ConteZero	Fratelli d'Italia ✓
Internazionale ✓	Tgcom24 ✓
Carlo Calenda ✓	Gruppo FI Camera ✓
Marco Castelnuovo ✓	Annagrazia Calabria
Davide Serra ✓	ForzaItalia ELEZIONI
Marco Cappato ✓	Luca Battanta
insopportabile ✓	angelo ra

Table 17: Following the approach of [78] we build the monopartite and directed network of retweets, in which an edge (i, j) indicates that j has retweeted i at least once. Then we select the directed subgraph generated by each community and here we report the list of the most retweeted users within the subgraphs of left- and right-leaning communities. The symbol ✓ next to the name indicates that the account has been verified.

Community ●	Community ●
Demian Yexil	GIOVANNI PANARIELLO
Salvatore Cos. marino29b	pulcedacqua
Stefjazz	Dome Buratti
A MARZO ABBIAMO IL DOVERE DI VOTARLI VIA	Beatrice Gallo
Maria	max
Luisa Loffredo	AriWelcome
Massimo	Ito
Serenella	c h i a r a ●
Libellula	Gio B.
Sergio Magugliani	Benevolence
Sherlock5Stelle	Marco Borrano
MAURIZIO	giuliana ●
Ori	Apudte
jan	Davide Segat ●
marinella	Quisque de populo
sonia oliva	Andrea Lisimberti
CELESTINO N.	Monica
Claudio Cocchi	Pio Bove
Elisabetta Prighel	ljsa
Vincenzo Lentini	Torakjkj #iosonoantifascista
tagliamoglilatestfivestars ✕	ariadne
ANDREA ZANETTIN	s.
	Debora

Table 18: Following the approach of [78] we build the monopartite and directed network of retweets, in which an edge (i, j) indicates that j has retweeted i at least once. Then we select the directed subgraph generated by each community and here we report the list of the most retweeting users within the subgraphs of M5S and information channels communities. The symbols ✓, ✕ and ● next to the name respectively indicate that the account has been verified, suspended or has not been found by the Twitter API.

Community ●	Community ●
Conci	ForzaItalia ELEZIONI
Daniele Dellavedova	Marcello Menna
Anna Iesu	GUI
carla martinelli	Forza Italia
Paolo Re	maria
vilma d'ippolito	Simona F.
ser -get-to	Alessandro Ritacco
Davide Scotti	SONIA GROTT
mamma orsa	Grazia
Roberto Locatelli	Nando
Mariarita Martegiani	Nando4cats
ranocchia	mariuccia frigerio
aleteja	Paolo Caminiti
Maurizio SÌ	Gruppo FI Camera
broncos 71	Forza Italia Online ✕
Amaro	Elena ✕
Angelina Scanu #antifascista #il4marzovotoPD	Forza Italia Roma
Annalisa	ILoveBlack
1N0 Qualunque	mari
Lia	FI Veneto.ELEZIONI
Brutus	patrizia gatto
bb	Il Mattinale
Miti Vigliero	Simone Furlan

Table 19: Following the approach of [78] we build the monopartite and directed network of retweets, in which an edge (i, j) indicates that j has retweeted i at least once. Then we select the directed subgraph generated by each community and here we report the list of the most retweeting users within the subgraphs of left- and right-leaning communities. The symbols ✓, ✕ and ● next to the name respectively indicate that the account has been verified, suspended or has not been found by the Twitter API.

Appendix C

This Chapter provides some theoretical results regarding the stochastic model presented in Chapter 4. First, in Sections C.1 and C.2, we describe the asymptotic behavior of the total number of features along time and we show some analytical findings regarding the asymptotic behavior of the mean number of edges in the actions-features bipartite network. Then, in Section C.3, we provide some statistical tools in order to estimate the parameters of the model. Finally, in the last Section C.4, we illustrate the employed data cleaning procedure.

C.1 Asymptotic behaviour of the total number of features

The random variable $L_t = \sum_{j=1}^t N_j$, that represents the total number of features present in the system at time-step t , has the following asymptotic behaviors as $t \rightarrow +\infty$:

- a) for $\beta = 0$, we have a logarithmic behavior of L_t , that is $L_t / \ln(t) \rightarrow \alpha$ almost surely;
- b) for $\beta \in (0, 1]$, we obtain a power-law behavior, i.e. $L_t / t^\beta \rightarrow \alpha / \beta$ almost surely.

The proof of these two statements is exactly the same as in [127], since the weights do not affect L_t .

C.2 Asymptotic behaviour of the mean number of edges

We here analyze the asymptotic behaviour of $\langle E_t \rangle = \mathbb{E}[E_t]$, where E_t is the total number of edges in the actions-features network at time-step t , that is the total number of ones in the matrix \mathbf{M} until time-step t . A first remark is that we have

$$E_t = \sum_{u=1}^t \sum_{c: T_c=u} k_{c,t}, \quad (\text{C.1})$$

where we denote by T_c the arrival time-step of feature c and

$$k_{c,t} = \sum_{j=1}^t m_{j,c} = 1 + \sum_{j=T_c+1}^t m_{j,c} \quad (\text{C.2})$$

is the degree of feature c at time-step t . Hence, we can write

$$\begin{aligned} \mathbb{E}[E_t | T_c \forall c \text{ with } T_c \leq t] &= \sum_{u=1}^t \text{card}(\{c : T_c = u\}) \mathbb{E}[k_{c,t} | T_c = u] \\ &= \sum_{u=1}^t N_u \mathbb{E}[k_{c,t} | T_c = u], \end{aligned} \quad (\text{C.3})$$

where we recall that N_u is $\text{Poi}(\lambda_u)$ -distributed with $\lambda_u = \alpha/u^{1-\beta}$. In the following subsections, we go further with the computations in the two extreme cases $\delta = 1$ and $\delta = 0$ since the behaviour for a general δ is a mixture of the two previous ones. A graphical representation of the evolution of $\langle E_t \rangle$ in the considered cases is provided in Figures 39 and 40, where the values are averaged over a sample of $R = 100$ simulations.

The case $\delta = 1$

In this case the inclusion probability of a feature c at time-step t simply is $p_{t,c} = \frac{1}{2}$. Therefore, since (C.2), we have

$$\mathbb{E}[k_{c,t} | T_c = t_c] = 1 + \frac{t - t_c}{2} \sim t/2.$$

Hence, by (C.3) and the above approximation, we can approximate $\langle E_t \rangle$ by the quantity

$$\frac{t}{2} \sum_{u=1}^t \lambda_u = \frac{\alpha t}{2} \sum_{u=1}^t w^{\beta-1} \sim \frac{\alpha t^{1+\beta}}{2\beta}. \quad (\text{C.4})$$

The case with $\delta = 0$ and the weights equal to a constant

Let us assume $\delta = 0$ and $w_{t,j,c}$ equal to a constant $w \in]0, 1]$ for all j, c, t , so that the inclusion probability of a feature c at time-step t is

$$p_{t,c} = \frac{k_{c,t-1}}{t} w.$$

Let us set $\langle k_{c,t} \rangle = \mathbb{E}[k_{c,t} | T_c = t_c]$ and observe that we have

$$\begin{aligned} \langle k_{c,t} \rangle &= 1 + w \sum_{\tau=t_c+1}^t \frac{\langle k_{c,\tau-1} \rangle}{\tau} \\ &= 1 + w \left[\sum_{\tau=t_c+1}^{t-1} \frac{\langle k_{c,\tau-1} \rangle}{\tau} + \frac{\langle k_{c,t-1} \rangle}{t} \right] \\ &= 1 + w \sum_{\tau=t_c+1}^{t-1} \frac{\langle k_{c,\tau-1} \rangle}{\tau} + \frac{w}{t} \left[1 + w \sum_{\tau=t_c+1}^{t-1} \frac{\langle k_{c,\tau-1} \rangle}{\tau} \right] \\ &= \left(1 + \frac{w}{t} \right) \left[1 + w \sum_{\tau=t_c+1}^{t-1} \frac{\langle k_{c,\tau-1} \rangle}{\tau} \right] \\ &= \dots \\ &= \left(1 + \frac{w}{t} \right) \left(1 + \frac{w}{t-1} \right) \dots \left(1 + \frac{w}{t_c+1} \right) \\ &= \frac{t_c!}{t!} \cdot (t+w) \cdot (t-1+w) \dots (t_c+1+w) \\ &= \frac{t_c!}{t!} \cdot \frac{(t+w) \cdot (t-1+w) \dots (t_c+1+w) \cdot (t_k+w) \dots (w+1) \cdot w}{(t_c+w) \dots (w+1) \cdot w}. \end{aligned}$$

Using the properties of the Γ -function, we can write

$$\langle k_{c,t} \rangle = \frac{t_c!}{t!} \frac{\Gamma(t+w+1)!}{\Gamma(t_c+w+1)!} = \frac{\Gamma(t_c+1)}{\Gamma(t+1)} \frac{\Gamma(t+w+1)!}{\Gamma(t_c+w+1)!} \sim \left(\frac{t}{t_c} \right)^w. \quad (\text{C.5})$$

Therefore, by (C.3) and the above approximation, we can approximate $\langle E_t \rangle$ by the quantity

$$\begin{aligned} \sum_{u=1}^t \lambda_u \frac{t^w}{u^w} &= \alpha t^w \sum_{u=1}^t u^{\beta-w-1} \\ &\sim \begin{cases} \alpha t^\beta \ln(t) & \text{if } w = \beta, \\ \frac{\alpha}{\beta - w} (t^\beta - t^w) \sim \frac{\alpha t^{\max\{w, \beta\}}}{|w - \beta|} & \text{if } w \neq \beta. \end{cases} \end{aligned} \quad (\text{C.6})$$

Remark: It is worthwhile to note that in the case of weights of the form $w_{t,j,c} = w_t$ for all j, c, t , where the random variables w_t take values in $[0, 1]$, are identically distributed with mean value equal to μ_w , and each of them is independent of all the past until time-step $t - 1$, we get for $\langle E_t \rangle$ the same asymptotic behavior as above, but with $w = \mu_w$.

The case with $\delta = 0$ and the weights depending only on c

Let us assume $\delta = 0$ and $w_{t,j,c} = w_c$ for all j, c, t , where the random variables w_c take values in $[0, 1]$, are independent and identically distributed with probability density function ρ , and each of them independent of the arrival time-step T_c of the feature. Moreover, we focus on the case $\beta < 1$, that is more interesting than the case $\beta = 1$. In this case the inclusion probability is

$$p_{t,c} = \frac{k_{c,t-1}}{t} w_c.$$

Using the same computations done above, we get

$$\mathbb{E}[k_{c,t}|T_c = t_c, w_c] \sim \left(\frac{t}{t_c}\right)^{w_c}$$

and so we can approximate $\mathbb{E}[k_{c,t}|T_c = t_c]$ by $\int_0^1 \left(\frac{t}{t_c}\right)^w \rho(w) dw$. Hence, using (C.3), we can approximate $\langle E_t \rangle$ by

$$\begin{aligned} \sum_{u=1}^t \lambda_u \int_0^1 \left(\frac{t}{u}\right)^w \rho(w) dw &= \int_0^1 t^w \sum_{u=1}^t \lambda_u u^{-w} \rho(w) dw = \\ &\alpha \int_0^1 t^w \sum_{u=1}^t u^{-(w-\beta+1)} \rho(w) dw = \alpha t^\beta \int_0^1 \frac{t^{w-\beta} - 1}{w - \beta} \rho(w) dw. \end{aligned} \quad (\text{C.7})$$

Therefore the asymptotic behavior of $\langle E_t \rangle$ depends on the asymptotic behavior of the above integral. In the sequel we analyze the case of the uniform distribution and the one of the “truncated” exponential distribution. To this purpose, we employ the Exponential integral

$$\text{Ei}(y) = - \int_{-y}^{+\infty} \frac{e^{-x}}{x} dx = \int_{-\infty}^y \frac{e^v}{v} dv,$$

which has the property $\lim_{y \rightarrow +\infty} \frac{e^y}{y \text{Ei}(y)} = 1$.

Example. *Uniform distribution on $[0, 1]$*

If $\rho(w) = 1, \forall w \in [0, 1]$ and equal to zero otherwise, we can compute the above integral and approximate $\langle E_t \rangle$ by

$$\begin{aligned} & \alpha t^\beta \left\{ \int_{-\beta \ln(t)}^{(1-\beta) \ln(t)} \frac{e^v}{v} dv - \int_{-\beta}^{1-\beta} \frac{1}{v} dv \right\} \\ &= \alpha t^\beta \left\{ \text{Ei}[(1-\beta) \ln(t)] - \text{Ei}[-\beta \ln(t)] + \ln\left(\frac{\beta}{1-\beta}\right) \right\} \end{aligned} \quad (\text{C.8})$$

Using the asymptotic properties of the Exponential integral, we find that the above quantity behaves for $t \rightarrow +\infty$ as

$$\frac{\alpha t}{(1-\beta) \ln(t)}.$$

Example. *Exponential distribution on $[0, 1]$*

If $\rho(w) = e^{1-w}/(e-1)$ for $w \in [0, 1]$ and equal to zero otherwise, the computation of the above integral leads to the approximation for $\langle E_t \rangle$ given by

$$\begin{aligned} & \frac{\alpha e^{1-\beta}}{(e-1)} t^\beta \left\{ - \int_{-\beta}^{1-\beta} \frac{e^{-x}}{x} dx + \int_{-\beta(\ln(t)-1)}^{(1-\beta)(\ln(t)-1)} \frac{e^v}{v} dv \right\} \\ &= \frac{\alpha e^{1-\beta}}{(e-1)} t^\beta \left\{ \text{Ei}[\beta] - \text{Ei}[-(1-\beta)] + \text{Ei}[(1-\beta)(\ln(t)-1)] - \text{Ei}[-\beta(\ln(t)-1)] \right\}. \end{aligned} \quad (\text{C.9})$$

Using the asymptotic properties of the Exponential integral, we find that the asymptotic behavior for $t \rightarrow +\infty$ of the above quantity is given by

$$\frac{\alpha t}{(e-1)(1-\beta) \ln(t)}.$$

C.3 Estimation of the model parameters

We here provide some statistical tools in order to estimate the parameters of the model introduced in Chapter 4.

The parameters α and β

The parameters α and β can be estimated using a maximum likelihood method, that is maximizing the probability to observe the given sequence of new features introduced in the system. More precisely, if we observe a number of T actions, we maximize the probability to observe $\{N_1 = n_1, N_2 = n_2, \dots, N_T = n_T\}$. Since all the random variables N_t are assumed independent Poisson distributed, we have

$$\begin{aligned}
 P(N_1 = n_1, \dots, N_T = n_T) &= P(N_1 = n_1) \prod_{t=2}^T P(N_t = n_t) \\
 &= P(N_1 = n_1 = \text{card}\{c : m_{1,c} = 1\}) \times \\
 &\quad \times \prod_{t=2}^T P(N_t = n_t = \text{card}\{c > L_{t-1} : m_{t,c} = 1\}) \\
 &= \text{Poi}(\alpha)\{n_1\} \prod_{t=2}^T \text{Poi}(\lambda_t)\{n_t\}.
 \end{aligned} \tag{C.10}$$

Hence, we choose as estimates the pair $(\hat{\alpha}, \hat{\beta})$ that maximizes function (C.10), or equivalently its log-likelihood expression

$$\ln(\text{Poi}(\alpha)\{n_1\}) + \sum_{t=2}^T \ln(\text{Poi}(\lambda_t)\{n_t\}).$$

The parameter δ

An estimate for the parameter δ is obtained maximizing the probability to observe the given actions-features matrix, i.e. maximizing the probability to observe the given biadjacency matrix rows $\{M_1 = m_1, M_2 =$

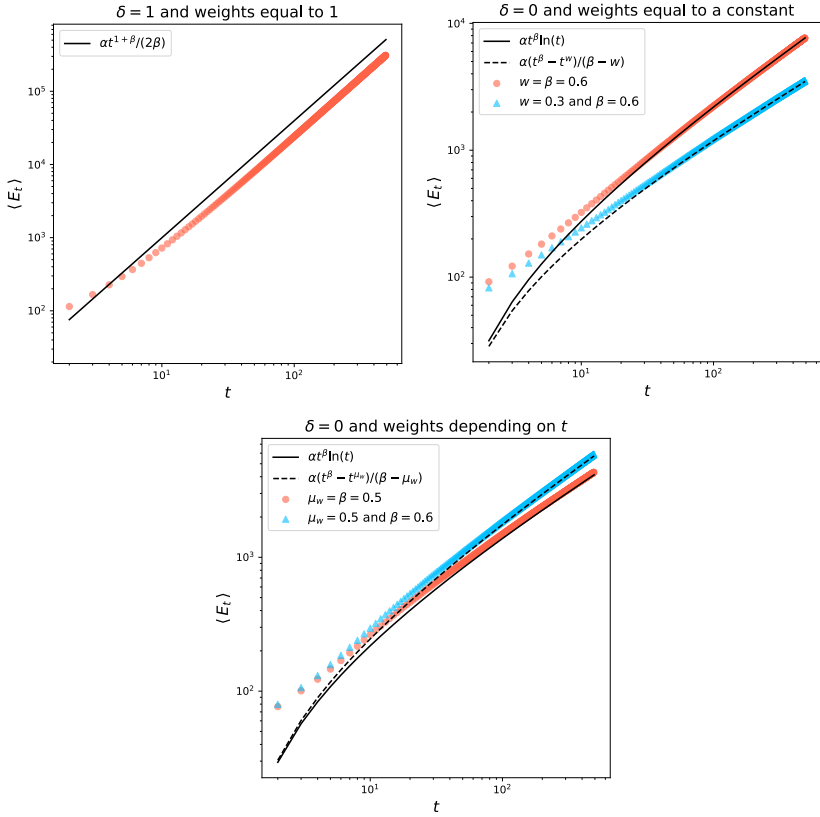


Figure 39: Evolution of $\langle E_t \rangle$, i.e. the mean number of edges along time. From top-left to bottom we have the cases: $\delta = 1$ and weights equal to 1; $\delta = 0$ and weights equal to a constant w (the blue triangles represent the case with $w \neq \beta$, while the red dots show the case $w = \beta$); $\delta = 0$ and weights depending only on t with uniform distribution on $[0, 1]$ (the blue triangles show the case with the mean value $\mu_w \neq \beta$, while the red dots describe the case with $\mu_w = \beta$). All simulations have been performed with $\alpha = 30$ and $\beta = 0.6$ (unless otherwise specified in the legend).

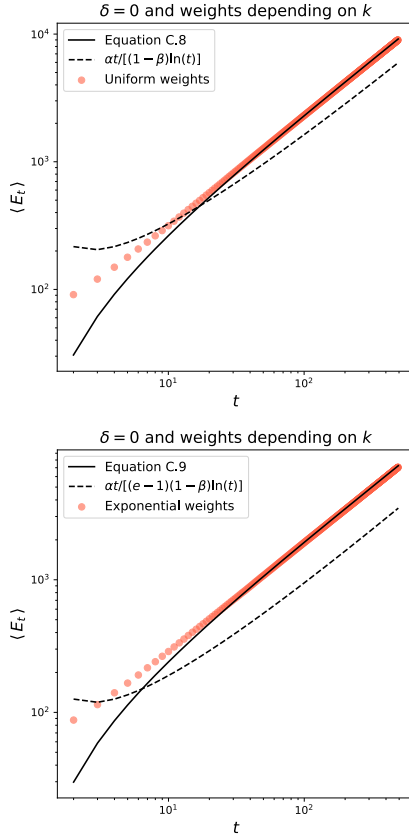


Figure 40: Evolution of $\langle E_t \rangle$, i.e. the mean number of edges along time. The Figure correspond to the case of $\delta = 0$ and weights depending only on c . Each panel considers a different distribution for the weights: uniform distribution (top panel) and truncated exponential distribution (bottom panel). The continuous lines refer to the values of the integrals (C.8) and (C.9), respectively, while the dashed lines show the final approximations. All simulations have been performed with $\alpha = 30$ and $\beta = 0.6$

$m_2, \dots, M_T = m_T\}$. More precisely, we have

$$\begin{aligned}
& P(M_1 = m_1, \dots, M_T = m_T) P(M_1 = m_1) \prod_{t=2}^T P(M_t = m_t | M_1, \dots, M_{t-1}) \\
& = P(N_1 = n_1 = \text{card}\{c : m_{1,c} = 1\}) \times \\
& \quad \times \prod_{t=2}^T P(M_{t,c} = m_{t,c} \text{ for } c = 1, \dots, L_{t-1}, \\
& \quad \quad N_t = n_t = \text{card}\{c > L_{t-1} : m_{t,c} = 1\} | M_1, \dots, M_{t-1}) \\
& = \text{Poi}(\alpha) \{n_1\} \prod_{t=2}^T \text{Poi}(\lambda_t) \{n_t\} \left\{ \prod_{c=1}^{L_{t-1}} p_{t,c}^{m_{t,c}} (1 - p_{t,c})^{1-m_{t,c}} \right\},
\end{aligned}$$

where $p_{t,c}$ and λ_t are defined in (4.3) and (4.4), respectively. Hence, we choose $\hat{\delta}$ that maximizes $P(M_1 = m_1, \dots, M_T = m_T)$. Since many terms in the previous equation do not depend on δ , the problem simplifies into the choice of the value of $\hat{\delta}$ that maximizes the following function

$$\prod_{t=2}^T \prod_{c=1}^{L_{t-1}} p_{t,c}^{m_{t,c}} (1 - p_{t,c})^{1-m_{t,c}} \quad (\text{C.11})$$

or, equivalently, taking the logarithm,

$$\sum_{t=2}^T \sum_{c=1}^{L_{t-1}} m_{t,c} \ln(p_{t,c}) + (1 - m_{t,c}) \ln(1 - p_{t,c}). \quad (\text{C.12})$$

It is worthwhile to note that the expression of the weights inside the inclusion probability (4.3) may possibly contain a parameter η . In this case, we maximize the above functions with respect to (δ, η) .

C.4 Data cleaning procedure

For the arXive and IEEE datasets, the data preparation procedure has been carried out using the Python package *NodeBox* [146] that allows to perform different grammar analyses on the English language. We use the library to categorise (as noun, adjective, adverb or verb) each

word in all title's or abstract's sentences, with the final purpose of selecting nouns and adjectives only. Then, all selected words are modified substituting capital letters with lowercases and transforming all plurals into singulars, again using the *NodeBox* package. Finally, we also remove special words such as "study", "analysis" or "paper", that may often appear in the abstract text but are not relevant for the description of the topic and for the purpose of our analysis. Authors names are similarly treated. Indeed, from each name we replace capital letters with lowercases and we modify it by considering only the initial letter for each reported name and the entire surname. To make an example, names such as "Peter Kaste" or "P. Jacob" are respectively transformed into "p.kaste" and "p.jacob". One drawback of this kind of analysis is that authors with more than one names who reported all of them or just some in different publications cannot be distinguished. Indeed, in this situation they would appear as distinct. For example "A. N. Leznov", "A. Leznov" or "Andrey Leznov" may probably identify the same person who reported respectively two initials, one initial or the full name in different papers. However, with this transformation they appear as two distinct authors, since they are respectively represented by the abbreviations "a.n.leznov" and "a.leznov". Despite this fact, no further disambiguation is performed on the names, since it would be computationally very expensive and beyond the purpose of our analysis.

References

- [1] Romualdo Pastor-Satorras, Alexei Vázquez and Alessandro Vespignani. Dynamical and Correlation Properties of the Internet. *Physical Review Letters*, 87(25):258701, 2001. 1, 14
- [2] Robert Meusel, Sebastiano Vigna, Oliver Lehmborg and Christian Bizer. The Graph Structure in the Web – Analyzed on Different Aggregation Levels. *Journal of Web Science*, 1(1):33–47, 2015. 1, 14
- [3] Réka Albert, Hawoong Jeong and Albert-László Barabási. Internet: Diameter of the world-wide web. *Nature*, 406:378–382, 2000. 1, 14
- [4] Michalis Faloutsos, Petros Faloutsos and Christos Faloutsos. On power-law relationships of the internet topology. In *ACM SIGCOMM computer communication review*, volume 29, pages 251–262. ACM, 1999. 1, 14
- [5] Alexei Vázquez, Romualdo Pastor-Satorras and Alessandro Vespignani. Large-scale topological and dynamical properties of the Internet. *Physical Review E*, 65(6):066130, 2002. 1, 14
- [6] Andrea Avena-Koenigsberger, Bratislav Misic and Olaf Sporns. Communication dynamics in complex brain networks. *Nature Reviews Neuroscience*, 19(1):17–33, 2017. 1
- [7] Rossana Mastrandrea andrea Gabrielli, Fabrizio Piras, Gianfranco Spalletta, Guido Caldarelli and Tommaso Gili. Organization and hierarchy of the human functional brain network lead to a chain-like core. *Scientific Reports*, 7(1):4888, 2017. 1
- [8] Guido Caldarelli. *Scale-Free Networks: Complex Webs in Nature and Technology*. Oxford University Press, Oxford, 2010. 1, 74
- [9] Brian Ball and Mark E J Newman. Friendship networks and social status. *Network Science*, 1(1):16–30, 2013. 2, 3

- [10] Mark E J Newman. The structure of scientific collaboration networks. *Proceedings of the National Academy of Sciences*, 98(2):404–409, 2001. 2, 151
- [11] Mark E J Newman. Clustering and preferential attachment in growing networks. *Physical review E*, 64(2):025102, 2001. 2
- [12] Mark E J Newman. Mixing patterns in networks. *Physical Review E*, 67(2):026126, 2003. 2
- [13] Albert-Laszlo Barabási, Hawoong Jeong, Zoltan Nédá, Erzsebet Ravasz, Andras Schubert and Tamas Vicsek. Evolution of the social network of scientific collaborations. *Physica A: Statistical mechanics and its applications*, 311(3-4):590–614, 2002. 2
- [14] Duncan J Watts and Steven H Strogatz. Collective dynamics of ‘small-world’ networks. *Nature*, 393(6684):440, 1998. 2
- [15] Mark E J Newman, Steven H Strogatz and Duncan J Watts. Random graphs with arbitrary degree distributions and their applications. *Physical review E*, 64(2):026118, 2001. 2
- [16] Mark E J Newman and Juyong Park. Why social networks are different from other types of networks. *Physical review E*, 68(3):036122, 2003. 2
- [17] Paul Erdős and Alfréd Rényi. On random graphs i. *Publicationes Mathematicae Debrecen*, 6:290–297, 1959. 13, 15
- [18] Paul Erdős and Alfréd Rényi. On the evolution of random graphs. *Publications of the Mathematical Institute of the Hungarian Academy of Sciences*, 5(1):17–60, 1960. 13, 15
- [19] Albert-László Barabási and Réka Albert. Emergence of Scaling in Random Networks. *Science*, 286(5439):509–512, 1999. 14, 17, 19, 34, 53
- [20] Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Reviews of modern physics*, 74(1):47, 2002. 13, 14, 15, 16, 17
- [21] Hawoong Jeong, Bálint Tombor, Réka Albert, Zoltan N Oltvai and Albert-László Barabási. The large-scale organization of metabolic networks. *Nature*, 407(6804):651, 2000. 14
- [22] Albert-László Barabási. Scale-free networks: A decade and beyond. *Science*, 325(5939):412–413, 2009. 14, 17
- [23] Mark E J Newman. Random graphs as models of networks. arXiv preprint cond-mat/0202208, 2002. 15

- [24] Béla Bollobás. Degree sequences of random graphs. *Discrete Mathematics*, 33(1):1–19, 1981. 16
- [25] George Udny Yule. A mathematical theory of evolution, based on the conclusions of Dr. JC Willis, FRS. *Philosophical Transactions of the Royal Society of London Series B*, 213:21–87, 1925. 21
- [26] Derek J De Solla Price. Networks of scientific papers. *Science*, pages 510–515, 1965. 21
- [27] Mark E J Newman. Power laws, pareto distributions and zipf’s law. *Contemporary physics*, 46(5):323–351, 2005. 21
- [28] Michael Golosovsky. Preferential attachment mechanism of complex network growth:” rich-gets-richer” or” fit-gets-richer”? arXiv preprint arXiv:1802.09786, 2018. 21
- [29] Ginestra Bianconi and Albert-László Barabási. Bose-einstein condensation in complex networks. *Physical review letters*, 86(24):5632, 2001. 21, 113, 120, 146
- [30] Ginestra Bianconi and Albert-László Barabási. Competition and multiscaling in evolving networks. *Europhysics Letters*, 54(4):436, 2001. 21, 23, 113, 120, 146
- [31] Guido Caldarelli, Andrea Capocci, Paolo De Los Rios and Miguel A Muñoz. Scale-free networks from varying vertex intrinsic fitness. *Physical review letters*, 89(25):258702, 2002. 21, 22, 23, 113, 120
- [32] Matúš Medo, Giulio Cimini and Stanislao Gualdi. Temporal effects in the growth of networks. *Physical Review Letters*, 107(23):238701, 2011. 23, 113, 122
- [33] Juyong Park and Mark E J Newman. Statistical mechanics of networks. *Physical Review E*, 70(6):066117, 2004. 23, 24, 70, 71, 75
- [34] Diego Garlaschelli and Maria I Loffredo. Maximum likelihood: Extracting unbiased information from complex networks. *Physical Review E*, 78(1):015101, 2008. 24, 70, 71, 75
- [35] Tiziano Squartini and Diego Garlaschelli. Analytical maximum-likelihood method to detect patterns in real networks. *New Journal of Physics*, 13, 2011. 24, 26, 28, 32, 33, 41, 57, 70, 71, 75
- [36] Fabio Saracco, Riccardo Di Clemente, Andrea Gabrielli and Tiziano Squartini. Randomizing bipartite networks: The case of the World Trade Web. *Scientific Reports*, 5(1):10595, 2015. 24, 32, 33, 34, 42, 57, 80, 168

- [37] Rossana Mastrandrea, Tiziano Squartini, Giorgio Fagiolo and Diego Garlaschelli. Enhanced reconstruction of weighted networks from strengths and degrees. *New Journal of Physics*, 16(4):043022, 2014. 24, 32, 33, 70, 75
- [38] Fabio Saracco, Mika J Straka, Riccardo Di Clemente, Andrea Gabrielli, Guido Caldarelli and Tiziano Squartini. Inferring monopartite projections of bipartite networks: an entropy-based approach. *New Journal of Physics*, 19(5):053022, 2017. 34, 42, 43, 54, 58, 64, 65, 69, 70, 71, 76, 81, 89, 151
- [39] Yu A Volkova. A Refinement of the Central Limit Theorem for Sums of Independent Random Indicators. *Theory of Probability and Its Applications*, 40(4):791–794, 1996. 35, 36, 54, 87, 88
- [40] Yili Hong. On computing the distribution function for the Poisson binomial distribution. *Computational Statistics & Data Analysis*, 59(1):41–51, 2013. 35, 36, 54, 87, 88
- [41] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society Series B*, 57(1):289–300, 1995. 36, 55, 88
- [42] James Bennett, Charles Elkan, Bing Liu, Padhraic Smyth and Domonkos Tikk. Kdd cup and workshop 2007. In *ACM SIGKDD Explorations Newsletter*, 9(2):51–52, 2007. 41
- [43] Rossana Mastrandrea, Tiziano Squartini, Giorgio Fagiolo and Diego Garlaschelli. Reconstructing the world trade multiplex: The role of intensive and extensive biases. *Physical Review E*, 90(6):062804, 2014. 41
- [44] Jasson D M Rennie and Nathan Srebro. Fast maximum margin matrix factorization for collaborative prediction. In *Proceedings of the 22nd International Conference on Machine Learning, ICML '05*, pages 713–719, New York, NY, USA, 2005. ACM. 42, 70
- [45] Shyam Boriah, Varun Chandola and Vipin Kumar. Similarity Measures for Categorical Data: A Comparative Evaluation. *Society for Industrial and Applied Mathematics - 8th SIAM International Conference on Data Mining 2008, Proceedings in Applied Mathematics 130*, vol. 1, pages 243–254, 2008. 43
- [46] Fan Chung and Linyuan Lu. Connected Components in Random Graphs with Given Expected Degree Sequences. *Annals of Combinatorics*, 6(2):125–145, 2002. 48
- [47] Jared M Diamond. Assembly of species communities, 1975. 53
- [48] Guido Caldarelli and Michele Catanzaro. The corporate Boards Networks. *Physica A: Statistical Mechanics and its Applications*, 338(1-2):98–106, 2004. 54

- [49] Michele Tumminello, Salvatore Miccichè, Fabrizio Lillo, Jyrki Piilo and Rosario N Mantegna. Statistically validated networks in bipartite complex systems. *PLoS One*, 6(3), 2011. 54
- [50] Stanislao Gualdi, Giulio Cimini, Kevin Primicerio, Riccardo Di Clemente and Damien Challet. Statistically validated network of portfolio overlaps and systemic risk. *Scientific Reports*, 6:39467, 2016. 54, 76
- [51] Navid Dianati. A maximum entropy approach to separating noise from signal in bimodal affiliation networks. arXiv preprint arXiv:1607.01735, 2016. 54
- [52] Mika J Straka, Guido Caldarelli and Fabio Saracco. Grand canonical validation of the bipartite international trade network. *Physical Review E*, 96(2):022306, 2017. 54, 69
- [53] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom and John Riedl. GroupLens : An Open Architecture for Collaborative Filtering of Netnews. In *Proceedings of the 1994 ACM Conference on Computer supported cooperative work*, CSCW '94, pages 175–186, 1994. 49, 55
- [54] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, 2008. 58, 81
- [55] Santo Fortunato. Community detection in graphs. *Physics Reports*, 486(3-5):75–174, 2010. 58, 81
- [56] Stanislav Sobolevsky, Riccardo Campari, Alexander Belyi and Carlo Ratti. General optimization technique for high-quality community detection in complex networks. *Physical Review E*, 90(1):012811, 2014. 58
- [57] Jonathan L Herlocker, Joseph A Konstan and John Riedl. Explaining collaborative filtering recommendations. In *Proceedings of the 2000 ACM Conference on Computer supported cooperative work*, CSCW '00, pages 241–250, 2000. 70
- [58] George Karypis. Evaluation of Item-Based Top-N Recommendation Algorithms. In *Proceedings of the 10th International Conference on Information and Knowledge Management*, CIKM '99, pages 247–254, 2001. 70
- [59] Federica Parisi, Guido Cardarelli and Tiziano Squartini. Entropy-based approach to missing-links prediction. *Applied Network Science*, 3(17), 2018. 71
- [60] Ruslan Salakhutdinov and Andriy Mnih. Probabilistic Matrix Factorization. In *Proceedings of the 20th Conference on Advances Neural Information Processing Systems*, NIPS '07, pages 1257–1264, 2007. 70

- [61] Yehuda Koren, Robert Bell and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, (8), pp.30-37, 2009. 70
- [62] Jiliang Tang, Charu Aggarwal and Huan Liu. Recommendations in Signed Social Networks. In *Proceedings of the 25th International Conference on World Wide Web, WWW'16*, pages 31–40, 2016. 70
- [63] Reinhard Diestel. *Graph Theory, 4th Edition* Springer, 2012.
- [64] Jean-Philippe Bouchaud and Antoine Georges. Anomalous diffusion in disordered media: Statistical mechanisms, models and physical applications. *Physics Reports*, 195(4):127–293, 1990. 74
- [65] David Wilkinson and Jorge F Willemsen. Invasion percolation: a new form of percolation theory. *Journal of Physics A: Mathematical and General*, 16(14):3365, 1983. 74
- [66] Dietrich Stauffer and Ammon Aharony. *Introduction to percolation theory: revised second edition*. Taylor & Francis, 1994. 74
- [67] Albert-László Barabási. The network takeover. *Nature Physics*, 8(1):14–16, 2011. 74
- [68] TNS opinion & social and Directorate-General Communications. Media use in the European Union. [Link here](#). 2018. 74
- [69] Sandra González-Bailón, Javier Borge-Holthoefer, Alejandro Rivero and Yamir Moreno. The dynamics of protest recruitment through an online network. *Scientific Reports*, 1:197, 2011. 74
- [70] Michael Conover, Jacob Ratkiewicz, Matthew R Francisco, Bruno Gonçalves, Filippo Menczer and Alessandro Flammini. Political polarization on twitter. In *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media, ICWSM '11*, pages 89–96, 2011. 74
- [71] Sandra González-Bailón, Javier Borge-Holthoefer and Yamir Moreno. Broadcasters and hidden influentials in online protest diffusion. *American Behavioral Scientist*, 57(7):943–965, 2013. 74
- [72] Lada A Adamic and Natalie Glance. The political blogosphere and the 2004 US election: divided they blog. In *Proceedings of the 3rd international workshop on Link discovery, LinkKDD '05*, pages 36–43, 2005. 74
- [73] Nicholas A Diakopoulos and David A Shamma. Characterizing debate performance via aggregated twitter sentiment. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '10*, pages 1195–1198, 2010. 74

- [74] Joseph DiGrazia, Karissa McKelvey, Johan Bollen and Fabio Rojas. More tweets, more votes: Social media as a quantitative indicator of political behavior. *PLoS One*, 8(11):e79449, 2013. 74
- [75] Marija Anna Bekafigo and Allan McBride. Who tweets about politics? Political participation of Twitter users during the 2011 gubernatorial elections. *Social Science Computer Review*, 31(5):625–643, 2013. 74
- [76] Adam Badawy, Emilio Ferrara and Kristina Lerman. Analyzing the Digital Traces of Political Manipulation: The 2016 Russian Interference Twitter Campaign. arXiv preprint arXiv:1802.04291. 74
- [77] Alexandre Bovet, Flaviano Morone and Hernán A Makse. Validation of Twitter opinion trends with national polling aggregates: Hillary Clinton vs Donald Trump. *Scientific reports*, 8(1):8673, 2018. 74
- [78] Alexandre Bovet and Hernán A Makse. Influence of fake news in Twitter during the 2016 US presidential election *Nature Communications*, 10(1):7, 2018. 74, 92, 96, 173, 174, 175, 176
- [79] Rachel K Gibson and Ian McAllister. Does cyber-campaigning win votes? Online communication in the 2004 Australian election. *Journal of Elections, Public Opinion and Parties*, 16(3):243–263, 2006. 74
- [80] Axel Bruns and Stefan Stieglitz. Quantitative approaches to comparing communication patterns on Twitter. *Journal of Technology in Human Services*, 30(3-4):160–185, 2012. 74
- [81] Gunn Sara Enli and Eli Skogerbø. Personalized campaigns in party-centred politics: Twitter and Facebook as arenas for political communication. *Information, Communication & Society* 16(5):757–774, 2013. 74
- [82] Javier Borondo, Alfredo J Morales, Juan C Losada and Rosa M Benito. Characterizing and modeling an electoral campaign in the context of Twitter: 2011 Spanish Presidential election as a case study. *Chaos: an interdisciplinary journal of nonlinear science*, 22(2):023138, 2012. 74
- [83] Guido Caldarelli, Alessandro Chessa, Fabio Pammolli, Gabriele Pompa, Michelangelo Puliga, Massimo Riccaboni and Gianni Riotta. A multi-level geographical study of Italian political elections from Twitter data. *PLoS One*, 9(5):e95809, 2014. 74
- [84] Michela Del Vicario, Sabrina Gaito, Walter Quattrociocchi, Matteo Zignani and Fabiana Zollo. Public discourse and news consumption on online social media: A quantitative, cross-platform analysis of the Italian Referendum. arXiv preprint arXiv:1702.06016, 2017. 74

- [85] Jacopo Bindi, Davide Colombi, Flavio Iannelli, Nicola Politi, Michele Sugarelli, Raffaele Tavarone and Enrico Ubaldi. Political Discussion and Leanings on Twitter: the 2016 Italian Constitutional Referendum. *arXiv preprint arXiv:1805.07388*, 2018. 74
- [86] Emilio Ferrara. Disinformation and social bot operations in the run up to the 2017 French presidential election. *First Monday*, 2017. 74, 75
- [87] Laura Cram, Clare Llewellyn, Robin Hill and Walid Magdy. UK General Election 2017: a Twitter Analysis. *arXiv preprint arXiv:1706.02271*, 2017. 74
- [88] Michela Del Vicario, Fabiana Zollo, Guido Caldarelli, Antonio Scala and Walter Quattrociocchi. Mapping social dynamics on Facebook: The Brexit debate. *Social Networks*, 50:6–16, 2017. 74
- [89] Walter Quattrociocchi, Guido Caldarelli and Antonio Scala. Opinion dynamics on interacting networks: media competition and social influence. *Scientific Reports*, 4:4938, 2015. 75, 76, 150
- [90] Michela Del Vicario, Gianna Vivaldo, Alessandro Bessi, Fabiana Zollo, Antonio Scala, Guido Caldarelli and Walter Quattrociocchi. Echo Chambers: Emotional Contagion and Group Polarization on Facebook. *Scientific Reports*, 6:37825, 2016. 75, 76, 150
- [91] Michela Del Vicario, Alessandro Bessi, Fabiana Zollo, Fabio Petroni, Antonio Scala, Guido Caldarelli, Eugene H Stanley and Walter Quattrociocchi. The spreading of misinformation online. *Proceedings of the National Academy of Sciences*, 113(3):554–559, 2016. 75, 76, 150
- [92] Ana Lucía Schmidt, Fabiana Zollo, Antonio Scala, Cornelia Betsch and Walter Quattrociocchi. Polarization of the vaccination debate on Facebook. *Vaccine*, 36(25):3606–3612, 2018. 75, 76, 150
- [93] Ana Lucía Schmidt, Fabiana Zollo, Antonio Scala and Walter Quattrociocchi. Polarization Rank: A Study on European News Consumption on Facebook. *arXiv preprint arXiv:1805.08030*, 2018. 75, 76, 150
- [94] Eytan Bakshy, Solomon Messing and Lada A Adamic. Exposure to ideologically diverse news and opinion on Facebook. *Science*, 348(6239):1130–1132, 2015. 75, 76
- [95] Dimitar Nikolov, Diego F M Oliveira, Alessandro Flammini and Filippo Menczer. Measuring Online Social Bubbles. *PeerJ Computer Science*, 1:e38, 2015. 75, 76
- [96] Emilio Ferrara and Zeyao Yang. Quantifying the effect of sentiment on information diffusion in social media. *PeerJ Computer Science*, 1:e26, 2015. 75

- [97] Ryota Kobayashi and Renaud Lambiotte. TiDeH: Time-Dependent Hawkes Process for Predicting Retweet Dynamics. In *Proceedings of the 10th International AAAI Conference on Weblogs and Social Media, ICWSM '16*, pages 191–200, 2016. 75
- [98] Lionel Tabourier, Anne Sophie Libert and Renaud Lambiotte. Predicting links in ego-networks using temporal information. *EPJ Data Science*, 5(1):1, 2016. 75
- [99] Onur Varol, Emilio Ferrara, Filippo Menczer and Alessandro Flammini. Early detection of promoted campaigns on social media. *EPJ Data Science*, 6(1):13, 2017. 75
- [100] Edwin T Jaynes. Information theory and statistical mechanics. *Physical review*, 106(4):620, 1957. 75
- [101] Tiziano Squartini, Iman van Lelyveld and Diego Garlaschelli. Early-warning signals of topological collapse in interbank networks. *Scientific Reports*, 3:3357, 2013. 75
- [102] Piotr Fronczak, Agata Fronczak and Maksymilian Bujok. Exponential Random Graph Models for Networks with Community Structure. *Physical Review E*, 88(3):032810, 2013. 75
- [103] Domenico Di Gangi, Fabrizio Lillo and Davide Pirino. Assessing systemic risk due to fire sales spillover through maximum entropy network reconstruction. *Journal of Economic Dynamics and Control*, 94:117–141, 2018. 75
- [104] Jeroen van Lidth de Jeude, Riccardo Di Clemente, Guido Caldarelli, Fabio Saracco and Tiziano Squartini. Reconstructing mesoscale network structures. *Complexity*, 2019. 75, 85, 86, 98, 150, 167
- [105] Tiziano Squartini and Diego Garlaschelli. *Maximum-Entropy Networks: Pattern Detection, Network Reconstruction and Graph Combinatorics*. Springer, 2017. 75, 85
- [106] Giulio Cimini, Tiziano Squartini, Fabio Saracco, Diego Garlaschelli, Andrea Gabrielli and Guido Cardarelli. The Statistical Physics of Real-World Networks. *Nature Reviews Physics*, 1(1):58–71, 2019. 75, 85
- [107] Pranav Dandekar, Ashish Goel and David Lee. Biased Assimilation, Homophily and the Dynamics of Polarization. *Proceedings of the National Academy of Sciences*, 110(15):5791–5796, 2013. 76
- [108] Seth Flaxman, Sharad Goel and Justin M Rao. Ideological Segregation and the Effects of Social Media on News Consumption. *SSRN Electronic Journal*, 1–39, 2013. 76

- [109] Mark E J Newman. *Networks. An introduction*. Oxford University Press, Oxford, 2010. 74
- [110] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *Science* (80-.), 286(5439):509–512, 1999. 113
- [111] Derek J De Solla Price. Networks of Scientific Papers. *Science* (80-.), 149(3683):510–515, 1965.
- [112] George Udny Yule. A Mathematical Theory of Evolution based on the Conclusions of Dr. J.C. Willis, F.R.S. *Journal of the Royal Statistical Society*, 88(3):433–436, 1925.
- [113] Michael Golosovsky. Preferential attachment mechanism of complex network growth: "rich-gets-richer" or "fit-gets-richer"? arXiv preprint arXiv:1802.09786, 2018.
- [114] Mark E J Newman. Power laws, Pareto distributions and Zipf's law. *Contemporary Physics*, 46(5):323–351, 2005.
- [115] Paul L Krapivsky and Sidney Redner. Organization of growing random networks. *Physical Review E*, 63(6):066123, 2001.
- [116] Paul L Krapivsky, Sidney Redner and Francois Leyvraz. Connectivity of growing random networks. *Physical Review Letters*, 85(21):4629–4632, 2000.
- [117] Sergey Dorogovtsev and José Fernando F Mendes. Evolution of reference networks with aging. *Physical Review E*, 62(2):021842, 2000.
- [118] Dashun Wang, Chaoming Song and Albert-László Barabási. Quantifying long-term scientific impact. *Science* (80-.), 342(6154):127–132, 2013. 122
- [119] Jean-Loup Guillaume and Matthieu Latapy. Bipartite graphs as models of complex networks. *Physica A: Statistical Mechanics and its Applications*, 371(2):795–813, 2006. 114
- [120] Mariano Beguerisse Díaz, Mason A Porter and Jukka Pekka Onnela. Competition for popularity in bipartite networks. *Chaos*, 20(4), 2010.
- [121] Angelo Mondaini Calvão, Crysttian Arantes Paixão, Flávio Codeco Coelho and Renato Rocha Souza. The consumer litigation industry: Chasing dragon kings in lawyer–client networks. *Social Networks*, 40:17–24, 2015. 114
- [122] Camille Roth and Jean-Philippe Cointet. Social and semantic coevolution in knowledge networks. *Social Networks*, 32(1):16–29, 2010. 114

- [123] Fabio Saracco, Riccardo Di Clemente, Andrea Gabrielli and Luciano Pietronero. From innovation to diversification: A simple competitive model. *PLoS One*, 10(11), 2015. 114
- [124] Stuart Kauffman. *At Home in the Universe: The Search for the Laws of Self-Organization and Complexity*. Oxford University Press, Oxford, 1996. 114
- [125] Paolo Boldi, Irene Crimaldi and Corrado Monti. A network model characterized by a latent attribute structure with competition. *Information Sciences*, 354:236–256, 2016. 114, 115
- [126] Etienne Birmelé. A scale-free graph model based on bipartite graphs. *Discrete Applied Mathematics*, 157(10):2267–2284, 2009. 114
- [127] Irene Crimaldi, Michela Del Vicario, Greg Morrison, Walter Quattrociocchi and Massimo Riccaboni. Modelling Networks with a Growing Feature-Structure. *Interdisciplinary Information Sciences*, 23(2):127–144, 2017. 114, 115, 120, 121, 128, 177
- [128] Patrizia Berti, Irene Crimaldi, Luca Pratelli and Pietro Rigo. Central limit theorems for an indian buffet model with random weights. *The Annals of Applied Probability*, 25(2):523–547, 2015. 118
- [129] Zoubin Ghahramani and Thomas L Griffiths. Infinite latent feature models and the indian buffet process. In *Advances in neural information processing systems*, pages 475–482, 2006. 118
- [130] Yee W Teh and Dilan Gorur. Indian buffet processes with power-law behavior. In *Advances in neural information processing systems*, pages 1838–1846, 2009. 118
- [131] Jure Leskovec, Jon Kleinberg and Christos Faloutsos. Graph Evolution: Densification and Shrinking Diameters. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(41), 2006. 128, 135
- [132] Emilio Ferrara, Roberto Interdonato and Andrea Tagarelli. Online Popularity and Topical Interests through the Lens of Instagram. In *Proceedings of the 25th ACM conference on Hypertext and social media*, HT '14, pages 24–34, 2014. 128, 140
- [133] Carolina Becatti, Guido Caldarelli and Fabio Saracco. Entropy-based randomization of rating networks. *Physical Review E*, 99(2):022306, 2019. xvi, i, 32, 75
- [134] Carolina Becatti, Guido Caldarelli Renaud Lambiotte and Fabio Saracco. Extracting significant signal of news consumption from social networks: the case of Twitter in Italian political elections, arXiv preprint arXiv:1901.07933, 2019. xvi, i

- [135] Carolina Becatti, Irene Crimaldi and Fabio Saracco. Collaboration and followership: a stochastic model for activities in bipartite social networks. arXiv preprint arXiv:1811.00418, 2018. xvi, i
- [136] The icons are courtesy of the website The Noun Project. Contributors of the icons: *Boy*, *Girl* and *Baseball Cap* by Moxilla, *Headphones* by Chad Remsing, *Smartphone* by Jonathan Lutaster, *Mouse* by See Link, *Radio* by Muhammad Riza, *Vintage Camera* by Setyo Ari Wibowo. 42
- [137] Actually, there is a tricky part. Since all users that were not author of retweeted posts have probability exactly equal to zero to spread a tweet, their probability to be the source of a directed V-motif is again exactly zero, for each of the possible tweet. Those cases were not considered in the FDR. 89
- [138] MovieLens source 49, 55, 71
- [139] Amazon dataset source 55, 71
- [140] Amazon dataset source 55
- [141] Anvur dataset source 56
- [142] All Music website 67
- [143] arXiv dataset for Theoretical High Energy Physics source 128
- [144] IEEE dataset for Automatic Driving source 128
- [145] Instagram dataset source 128
- [146] Python package NodeBox website 185



Unless otherwise expressly stated, all original material of whatever nature created by Carolina Becatti and included in this thesis, is licensed under a Creative Commons Attribution Noncommercial Share Alike 2.5 Italy License.

Check creativecommons.org/licenses/by-nc-sa/2.5/it/ for the legal code of the full license.

Ask the author about other uses.