The Effect of Friend Recommenders on the Formation of Echo Chambers and Polarization in Social Networks

Miguel Ángel Sánchez Cortés*

Social Networks & Online Markets, MSc. in Data Science, Sapienza University of Rome

(Dated: October 17, 2024)

In this project we aim to reproduce and extend the Cinus et al. (2022)'s work on assessing the effect of friend recommendation algorithms on the formation of echo chambers and polarization on social networks and on the exacerbation of the popularity bias and triadic closure effects. In order to do this, we propose a modified version of the PROD methodolody to test the significance on such effects on a random network model interacting with an opinion dynamics model. We found similar results as the ones presented by Cinus et al. (2022), by observing that all studied recommendation algorithms increase popularity bias and have a significant effect on increasing echo chamber behavior and polarization for low-modular and highly homophilic network structures. We also test different idealized intervention policies in order to try to mitigate this effect.

Keywords: social networks, echo chambers, polarization

I. INTRODUCTION

Online social networks have become an integral part of how people interact with each other in their everyday life. It has been shown that thanks to these interactions many emergent phenomena arise, such as the so-called *echo-chamber effect* (Quattrociocchi et al. 2016) due to which users tend to interact with other users with the same ideological leaning, reinforcing their own viewpoints, and thus getting progressively more polarized opinions. However, it has been suggested that emergent phenomena in online social networks is not only governed by well-known mechanishms such as homophily (McPherson et al. 2001) or preferential-attachment (Barabási and Albert 1999), but it is also shaped by recommender systems, algorithms that suggest new content or connections among the users of a social media platform. In fact, it has been shown that recommender systems not only play a significant role on the formation of echo chambers (Cinelli et al. 2021) and polarization (Dandekar et al. 2013), but are also prone to reinforcing popularity bias (Abdollahpouri 2019) and highlighting inequality and inequity on minority groups (Espín-Noboa et al. 2022). Due to these factors, and the reluctance of social media platforms to contribute in mitigating the impact of their own algorithms (Bagchi et al. 2024), studying algorithmic recommenders on social networks has become crucial to understand the extent of the effects of such recommenders and the mechanisms behind this effects, with the objective of auditing social-media platforms and building better policies that aim to reduce such effects.

In this project, we aim to reproduce and extend the results obtained by Cinus et al. (2022), studying the effect of friend recommenders, algorithms that recommend users to other users on a social media platform, on the formation of echo chambers and polarization. Given the difficulty of performing field experiments to obtain data on people recommenders acting on social media platforms, Cinus et al. (2022) propose a simulation approach by studying the effect of friend recommenders on well-known opinion dynamic models that interact also with a real-world-like network structure. The outline of this approach can be summarized as follows:

(1) Defining a real-world network structure model that incorporates homophily and community structure such that it resembles the local network of a group of friends on a social media platform and initialize each friend with an initial opinion, where two friends are *similar* if they have similar opinions.

(2) Defining an Opinion Dynamics model that will govern the way how these friends interact through the network and change their opinions after interactions.

(3) Defining a recommender system that suggests links to a given node to interact with, this can simulate the interaction of users on a social media platform and how opinions evolve under the influence of friend recommendations.

(4) Performing a statistical test that quantifies if the difference between a given measure obtained in the network with and without a recommendation system is statistically significant. Specifically, Cinus et al. (2022), define measures that quantify the echo chamber effect and polarization on networks.

* Email address: sanchezcortes.2049495@studenti.uniroma1.it

In this project, we reproduce and extend Cinus et al. (2022)'s work by assessing the effect of people recommenders not only on echo chambers and polarization, but on *triadic closure* (Granovetter 1973) and *popularity bias*. We also study the effect of the node2vec (Grover and Leskovec 2016) algorithm as a friend recommender by presenting the changes in network structure and given measures after applying this algorithm to different initial conditions. This last analysis is done qualitatively due to lack of computational power but a quantitative approach is left for future work.

II. METHODOLOGY

In this section we outline the methodology proposed by Cinus et al. (2022) and therefore, the methodology followed by this project. First, we will present the framework through which the network nodes will interact via an opinion dynamic model and a given recommender system, defined as a modified version of the PROD framework presented by Cinus et al. (2022). We then, will present an overview of the real-world-like network model used to perform the simulations, along with the chosen opinion dynamics model and the friend recommendation algorithms incorporated to the interaction framework. Finally, we will present how to use the modified PROD in order to evaluate the effect of a recommender algorithm on a given metric via a Monte-Carlo simulation procedure and performing hypothesis testing.

A. Interaction Framework

Cinus et al. (2022) define PROD as an interaction framework between a directed social network $G = (V, E, O)$, where $(u, v) \in E$ indicates that "u follows v" and $O : V \to [0, 1]$ represents a function that assigns each node to an opinion, and a link recommender algorithm $l_G : V \times V \to [0, 1]$, that assigns a recommended node $v \in V$ to a given node $u \in V$ with a probability $p_v \in [0, 1]$ given by the recommendation strength of each recommender. On top of these two main inputs, PROD adds an opinion dynamics model (ODM) that defines an *update* rule that modifies the opinions of each node when they interact with each other. In this project we propose a modified version of the PROD framework given by Algorithm 1 by making a simplification and assuming that at each recommendation step, a node u always accepts the given recommended node v. This choice is motivated by previous literature (Ferrara et al. 2022), since it has been shown that the employed acceptance policy effect is negligible in shaping the network structure (Fabbri et al. 2021).

Algorithm 1: Modified PROD

```
Input : Network with opinions G = (V, E, O),
        Friend recommender l_G : V × V → [0, 1],
Data  : Interactions per step S ∈ ℕ⁺,
        Max. recommendations R_max ∈ ℕ,
        Time steps T_max ∈ ℕ⁺
r, t ← 0
α ← R_max / ((T_max / 2) · S · |V|)
while t < T_max do
    t++
    σ ← random permutation of V
    forall u ∈ σ do
        s ← 0
        while s < S do
            if Bernoulli(α) then
                v ← l_G(u)
                w ← random node from N_out(u)
                E ← E \ {u, w}
                E ← E ∪ {u, v}
                UpdateRule(O_u, O_v)
                s++
                r++
                if r = R_max then
                    α ← 0
                end
            else
                v ← random node from N_out(u)
                UpdateRule(O_u, O_v)
                s++
            end
        end
    end
end
```

Pseudocode Overview: According to Algorithm 1, the modified PROD takes as an input a network G with assigned opinions O and a given friend recommender algorithm l_G. For a number of time steps T_{max}, nodes in the network are visited in random order. At each time step, every node interacts with S other nodes via two different mechanisms:

(1) In the first case, the recommender algorithm returns a node v, as a recommendation for node u and a directed link (u, v) is created for this recommendation. Afterwards, both nodes interact through the ODM. It is important to note that, for every addition, a random link with a neighbor is also removed. This is motivated by Dunbar's number (Dunbar 1992), and the social theory behind it that states that people have a limit on the number of ties that they can maintain active in time. This also ensures that the network density is kept constant in order to attribute any particular effect to the recommendation algorithms rather than a change in density.

(2) In the second case, the given node u interacts with a random neighbor v by updating its opinion following the ODM.

When a maximum number of recommendations R_{max} is given, the recommender stops and the interactions are given solely by the ODM in order to let emerge long-term effects on the opinion distribution. Cinus et al. (2022) also define an internal parameter α in order to balance the number of steps between the two previous cases (friend recommendation along with ODM and pure ODM). This ensures that the simulations with different recommendation algorithms give the same amount of recommendations R_{max} and have the same number of interactions between nodes ($S \cdot |V| \cdot T_{max}$), with each one of the cases mentioned before lasting approximately $\frac{T_{max}}{2}$ steps.

In the next subsections, we will introduce and discuss the random network model used to test recommendation algorithms using PROD, along with its two main ingredients: the ODM and the friend-recommendation algorithms. Finally, we will introduce Cinus et al. (2022)'s method to evaluate the effect of friend recommenders on different network metrics and the metrics used for this project.

B. Random Network Model

In order to measure the effect of friend-recommenders in social networks, specifically their contribution on the formation of echo chambers and polarization, it is necessary to build a random network model such that it allows us to measure these phenomena in the network with and without the recommender. At the same time, this network model should be realistic in comparison with real-world social networks. With this framework in mind, we propose Cinus et al. (2022)'s extension of the LFR-Benchmark model (Lancichinetti et al. 2008). This model allows for the creation of real-like random networks by tuning the number of nodes N of the network and two main parameters:

(1) Homophily ($0 \leq \eta \leq 1$): A parameter describing how close is the opinion of a node to the opinion of its local community (on average). In general, the higher the value of η, the higher the agreement of opinion in nodes of the same community.

(2) Modularity ($0 \leq \mu \leq 1$): A parameter describing the ratio of inter-community edges. In general, the lower the value of μ, the higher the segregation of the network in different communities.

The first parameter is introduced by Cinus et al. (2022) in order to control the echo-chamber level of the local graph of a node in a social network, and the second parameter is a parameter of the original LFR-Benchmark model. The procedure to create a random network using this model outputs a directed network G and the vector of opinions of its nodes $O \in [0,1]^N$. This is done by performing two main steps:

(1) Given N the number of nodes and μ the modularity parameter, we obtain a set of edges E and a partition of nodes into communities $c : V \to C$ by creating a network using the original LFR-Benchmark model.

(2) Given the previous network and the homophily parameter η, we generate node opinions by picking uniformly at random an opinion o_k for each community $k \in C$ by drawing $o_k \sim U(0,1)$. Then, the opinion of a node $v \in V$ is obtained by performing a Bernoulli trial with probability ν. If the trial is successful, the node takes the opinion of its community; otherwise, it generates an opinion uniformly at random.

From the previous procedure we can notice that, since the network structure is created entirely by the LFR-Benchmark model, it mantains the same real-world network traits as this model such as a power-law degree distribution (see Appendix A) and a power-law community size distribution (see Lancichinetti et al. (2008)). Therefore, this procedure allows us to generate real-like random networks with opinions and community structure. A network with high values of μ and ν would have a polarized structure, with a strong community structure and nodes in each community sharing the same opinions. On the other side, for low values of both parameters, it would have no discernible community structure and opinions would be distributed almost at random. In Figure 1 we present some examples of generated networks for limit cases in the parameter space.

C. Opinion Dynamics Model

In order to update the opinions of the nodes in the network generated by using the model mentioned above, we proposed a very widespread model of opinion dynamics: the Bounded Confidence Model (Deffuant et al. 2000). This model assumes that opinions are equivalent and that interactions between nodes happen when people have close opinions, updating the resulting opinions to a an opinion closer to each others. This model can be thought as the model of a political debate, for example, where agents with similar opinions within a

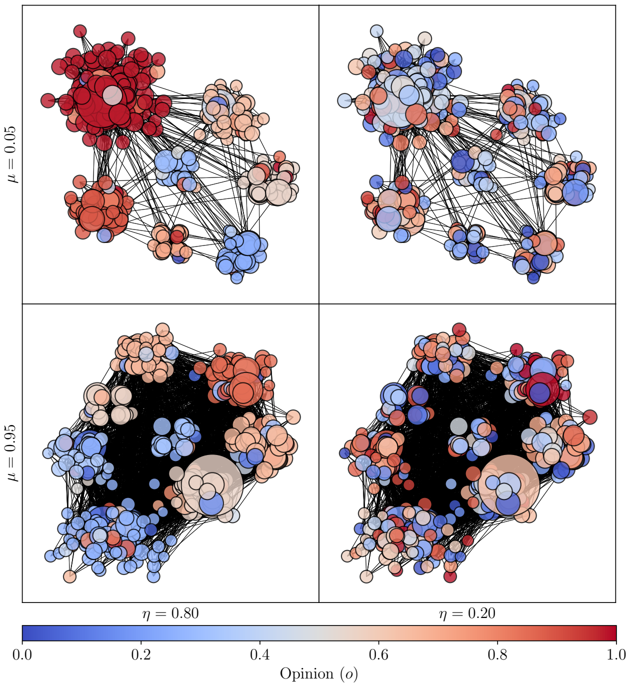


Figure 1: Examples of generated networks with opinions obtained with the extended LFR-Benchmark Model for four limit cases of the parameter space (homophily in the x-axis and modularity in the y-axis). Colors correspond to each node's opinions and sizes are proportional to each node's degree. Each community is assigned a particular point in space for better visibility.

range reach an agreement, and agents with opinions far from each others don't bother to interact. Formally, in the Bounded Confidence Model (BCM), whenever two nodes interact, their opinions are modified only when they are within a confidence interval $\epsilon \in [0,1]$ from each other. If they are, and they are in a directed network, when u interacts with v, u moves closer to v's opinion. Mathematically, the update rule at time $t+1$ is defined by:

$$o_u^{t+1} = \begin{cases} o_u^t + \mu_c \cdot (o_v^t - o_u^t), & \text{if } |o_u^t - o_v^t| < \epsilon \\ o_u^t, & \text{otherwise} \end{cases},$$

where μ_c represents the strength of the interaction and is a parameter defined a priori. This is a simple yet powerful model with complex behaviour since it can exhibit polarization of opinions in some cases. In this project we considered it is enough to capture the nature of online debate although in future work it could be better to use more complex models that capture other phenomena like the backfire effect Lewandowsky et al. (2017), for example.

D. Friend Recommendation Algorithms

Friend recommendation algorithms are used in most of modern social media platforms. They can mainly be categorized by means of the input information they use to make predictions (Guy 2018): network structure information (e.g. friends-of-friends recommendations) or content, activity and behavioural information (e.g. similar interests recommendation). Given that we can only obtain the network structure and opinions from our random network model, in this project we study three link prediction algorithms that use only the network structure as input —Personalized PageRank (Kumar et al. 2020), Who-To-Follow (Gupta et al. 2013), and Node2Vec (Grover and Leskovec 2016) — a method who only relies on nodes opinions, and a random recommender to act as a baseline.

Personalized PageRank (PPR). This method is based on the obtention of the PageRank measure of centrality (Page et al. 1999). Let $\mathbf{A}$ be the adjacency matrix of a directed network $G = (V, E)$, where $A_{ij} = 1$ if $(i, j) \in E$, and $\mathbf{P}$ be its column-stochastic version, where P_{ij} represents the probability of a random walker to transition from i to j. The PageRank vector $\mathbf{x}$ is defined as the solution to the eigenvalue problem (Gleich 2014):

$$\left(\alpha_p P + (1-\alpha_p)\mathbf{v}\cdot\mathbf{e}^T\right)\mathbf{x} = \mathbf{x},$$

where $\mathbf{e}$ is a vector of ones, $\alpha_p \in [0,1]$ is known as the *damping factor* and represents the probability that a random walker in the network will follow the edges of the network according to the transition matrix $\mathbf{P}$ instead of teleporting to another random node. On the other side, $1-\alpha_p$ gives the probability that a random walker will teleport to a random node selected from the probability distribution $\mathbf{v}$. In other words, the PageRank vector $\mathbf{x}$ is the stationary distribution of the Markov chain given by the above matrix $M := \left(\alpha_p P + (1-\alpha_p)\mathbf{v}\cdot\mathbf{e}^T\right)$.

The i-th entry of the vector $\mathbf{x}$ gives the PageRank of node i, representing the probability that a random walker ends up in node i when starting at random in the network. This value gives a measure of importance of a node in a network and it can also define a measure of importance of a node in a network from the point of view of another node. Given $\mathbf{v} = \mathbb{1}_i$, the $|V|$-dimensional vector with all zeros and 1 at the i-th entry, we define the Personalized PageRank score as the solution of the eigenvalue problem defined above. This modification allows us to compute a random walk biased to starting from node i and the scores given for all the other nodes can be used to recommend other nodes based on its network structure.

Who-To-Follow (WTF). The Who-To-Follow service (Gupta et al. 2013) represents X's (formerly Twitter) friend recommendation algorithm. This algorithm is based on the Stochastic Approach for Link-Structure Analysis (SALSA) algorithm proposed by Lempel and Moran (2001), originally developed for web search. The original SALSA algorithm constructs a bipartite network from a collection of webpages (called *hubs* and *authorities* in each side) and performs two distinct random walks, where each one visits only nodes from one side of the bipartite network by traversing two edges: one forward and one backward.

Mathematically, given a node $u \in V$ from a network $G = (V, E)$, the SALSA algorithm builds a bipartite network $G_b = (V_h, V_a, E_b)$, where V_h is the set of hub nodes and V_a is the set of authority nodes, i.e. the nodes the hubs follow on the original network. Let $\mathbf{A}$ be the $|V_a| \times |V_h|$ matrix with $A_{ij} = 1$ if $(i_h, j_a) \in E_b$ and zero otherwise. Let $\mathbf{M}$ be the column-stochastic version of $\mathbf{A}$, then we can write the random walks in terms of their transition matrices as in the PageRank case as follows:

$$\begin{cases} \mathbf{r} = & \mathbf{M}\mathbf{s} \\ \mathbf{s} = & d \cdot \mathbb{1}_u + (1-d) \cdot \mathbf{M}^T \mathbf{r} \end{cases},$$

where d is the equivalent of the *damping factor* and $1 - d$ controls the probability of teleporting the random walker to the node u as in the PPR case. The equations above are initialized with $s_u = 1$ and zero otherwise, and recursively applied until the vectors $\mathbf{r}$ and $\mathbf{s}$ converge to their stationary distribution. These vectors represent the relevance scores of authorities and similarity scores of hubs respectively.

In the WTF case, the hubs side of the bipartite network defined above is populated with node's *u circle of trust*, computed performing the PPR algorithm on this node and selecting the top-k nodes obtain. Afterwards, the authorities side of the network is populated with nodes that the hubs follow. Finally, the SALSA algorithm is performed, which assign scores to both sides, and the list of node recommendations for node u is defined as the nodes V_a sorted by the values in $\mathbf{r}$.

Node2Vec. The node2vec algorithm (Grover and Leskovec 2016) is one of the most popular node embedding algorithms, which maps nodes in a network to a low-dimensional space of features by maximizing the likelihood of preserving the nodes neighborhoods. This algorithm is inspired by the word2vec algorithm proposed by Mikolov et al. (2013) and its use of the skip-gram model to learn a vector embedding of a word such that neighboring words that give context to the word have similar embeddings.

In broad terms, the node2vec algorithm obtains node embeddings from a network by performing multiple biased random walks on it and exploring each node's neighborhoods. The length the number of the random walks are hyperparameters, along with the return parameter r, controlling the likelihood of revisiting a previous node in the walk, and the in-out parameter q, controlling the of the walk to stay on a local neighborhood or exploring the network as a whole. After generating the random walks, the node2vec algorithm treats the sequences of nodes visited as "sentences" in the word2vec sense and uses the skip-gram model to learn node embeddings by maximizing the probability of observing a node's neighbors in the walk given the current node. These embeddings are then optimized so that nodes that are closer in the network have similar embeddings.

In this project, we use node2vec to recommend to a node u, its most similar node in the embedding space according to the *cosine similarity* of their embedding vectors. Mathematically, given node u, a recommended node v, and their respective embedding vectors $\mathbf{u}_p$ and $\mathbf{v}_p$, the cosine similarity of nodes u and v is given by:

$$\text{CosSim}(u, v) = \frac{\mathbf{u}_p \cdot \mathbf{v}_p}{||\mathbf{u}_p||||\mathbf{v}_p||},$$

where $\text{CosSim}(u, v) \in [-1, 1]$, with $\text{CosSim}(u, v) = 1$ indicating maximum similarity, $\text{CosSim}(u, v) = 0$ indicating no similarity, and $\text{CosSim}(u, v) = -1$ indicating maximum dissimilarity.

Opinion-biased algorithm (OBA). In contrast with traditional network-based friend recommendation algorithms, Cinus et al. (2022) propose an opinion-biased recommender inspired by the biased bounded confidence model developed by Sîrbu et al. (2019). This algorithm simply recommends nodes based on the similarity of opinions between them. Formally, given nodes u and v and their respective opinions $o_u, o_v \in [0, 1]$, we define their opinion distance as the difference between their opinions $d_{uv} = |o_u - o_v|$. Therefore, we can define a recommendation score between both nodes as:

$$p_{uv} = \frac{d_{uv}^{-\gamma}}{\sum_{(i,j) \in E} d_{ij}^{-\gamma}},$$

where γ is a constant of similarity importance: if γ is high, the recommendation score for very similar nodes will be higher.

Random Algorithm. In order to have a baseline idealized recommender to compare our results to, in this project we also define a *random* recommender algorithm, that simply recommends for a given node u, a node v sampled uniformly at random from the node set V.

It is important to notice that every one of the recommender algorithms defined above (except the random recommender) defines a given *measure* of importance to a recommendation that is not always within the same range as the other ones and more importantly, that is not always sampled from the same probability distribution. To avoid this issue, we simply rank the recommendations by the value given by the algorithm and recommend the top-1 node every time (with exception of the node neighbors, since we can't recommend an already existing node) following the modified-PROD algorithm (see Algorithm 1).

E. Monte Carlo Evaluation Method

In this subsection, we present Cinus et al. (2022)'s methodology to evaluate the effect of a recommendation algorithm on a given metric m of a specific phenomenon, e.g. echo chambers, polarization, triadic closure, etc. using the modified-PROD algorithm. Their proposed evaluation methodology performs the following steps:

(1) Define a grid of different values of the parameters μ, η and for each pair of parameters, generate a sequence $\mathcal{G}_{\mu,\eta}$ of K random networks with opinions, with different values of modularity and initial homophily (see the Random Network Model subsection II B).

(2) On each network $G \in \mathcal{G}_{\mu,\eta}$, run the modified-PROD algorithm without any recommendation system i.e., with $R_{max} = 0$, obtaining a network G_0. Then, run the modified-PROD algorithm with a recommendation algorithm l, obtaining a network G_l.

(3) Given a metric m for a given phenomenon (see Metrics subsection below), obtain $m(G)$, $m(G_0)$, $m(G_l)$.

(4) Obtain the average difference Δm between the metric values with and without recommender:

$$\Delta m = \frac{1}{K} \sum_{G \in \mathcal{G}_{\mu,\eta}} m(G_l) - m(G_0)$$

(5) Perform a *Kolmogorov-Smirnoff* test to compare the distribution of the Δm and the distribution of the average difference Δm_0 between the initial metric value and the metric value without recommender. This test determines if the observed effect is statistically significant or not.

F. Metrics

In the following subsection we are going to present the metrics used in this project in order to assess the effect of friend recommenders on the formation of: echo chambers, polarization, triadic closure and popularity bias. In this project we proposed a metric to measure each one of these phenomena individually, in accordance to what is proposed by Cinus et al. (2022).

Clustering Coefficient. The clustering coefficient is a widespread network metric that quantifies the tendency of nodes in a social network to form *cliques*. This measure also allows us to verify if the recommendations given by our algorithms are making the network more clustered by closing the triangles between pairs of nodes, giving rise to triadic closure in the network. Mathematically, the clustering coefficient for a node u is defined for directed networks as follows (Fagiolo 2007):

$$c_u = \frac{2}{k_u^{tot}(k_u^{tot} - 1) - 2k_u^{\leftrightarrow}} T_u,$$

where T_u is the number of directed triangles that pass through node u, k_u^{tot} is the sum of the in and out degrees of node u, and $k_u^{\leftrightarrow}$ is the reciprocal degree of u. From this coefficient, we can also define the average clustering coefficient of the network by taking the mean across all nodes.

In-Degree Gini Coefficient. The Gini coefficient (Gini 1912) is a statistical measure of income inequality within a population. In the context of inequality on networks, Ferrara et al. (2022) proposed to measure the Gini coefficient of the in-degree distribution of a network to measure *popularity bias* and assess whether algorithmic recommendations exacerbate this bias. Mathematically, the Gini coefficient of the in-degree distribution π^{in}, sorted in ascending order, is defined as:

$$\text{Gini}(\pi^{in}) = \frac{\sum_{i=1}^{n}(2i - n - 1)\pi_i^{in}}{n \sum_{i=1}^{n} \pi_i^{in}},$$

where the higher the Gini coefficient, the more skewed and unequal is the in-degree distribution across all nodes.

Neighbors Correlation Index (NCI). The NCI, defined by Cinus et al. (2022), is defined simply as the Pearson correlation coefficient $\rho(\mathbf{o}, \mathbf{o}^N) \in [-1, 1]$ between the opinion vector of all the nodes in the network $\mathbf{o}$, and the vector containing the average opinion of each node's neighbors, where each entry is given by:

$$\mathbf{o}_i^N = \frac{1}{|N(i)|} \sum_{j \in N(i)} o_j,$$

where $N(i)$ is the set of neighbors of node i and o_i the opinion of node i. An NCI value of -1 means perfect anticorrelation: each node has the opposite opinion of its neighbors. On the other side, 1 represents the emergence of the perfect echo chamber: each node has the same opinion of all of its neighbors.

Random Walk Controversy Score (RWC). As a polarization metric, in this project we use the RWC proposed by Garimella et al. (2018). Given two disjoint components X and Y of the network, the RWC is mathematically defined as:

$$\mathrm{RWC} = P_{XX}P_{YY} - P_{XY}P_{YX},$$

where P_{ij} is the probability for a random walker that ends in partition j to have started in partition i. We can observe that $\mathrm{RWC} \in [-1, 1]$ and that it is close to 1 when the probability of crossing the partition is low, and close to 0 when the probability of crossing sides is the same with respect to staying in the same partition side. It is important to notice that this measure is not skewed by the size of the partition since the probability of starting from each partition is the same, at the same time, it is not skewed by the total degree of the nodes, since the probability is conditional on ending on each partition. This metric can be seen as the probability that a random user is exposed to content from the opposite side of its opinion spectrum.

In the case of this project we define the partition X as the set of nodes with opinions $o_i < 0.5$ and the partition Y as the opposite case. In practice we use the PPR algorithm defined above to perform the random walks starting from each one of the partitions and in the same way as Cinus et al. (2022), we prune high-degree nodes (over the 95th percentile of the degree distribution) in order to force a restart of the random walk when arriving to highly attractive nodes.

III. RESULTS

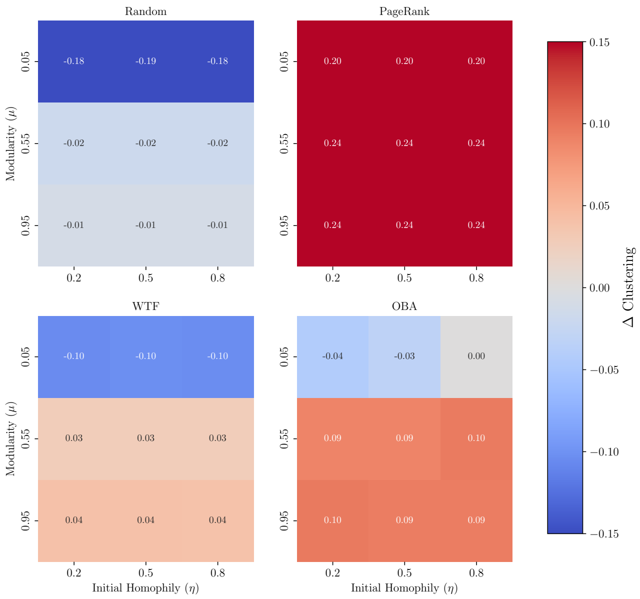
In this section we present the results obtained by using PROD (see Algorithm 1) on a social network modeled by the random network model presented in subsection II B and the ODM presented in subsection II C to evaluate the effect of the recommendation algorithms presented in subsection II D on the formation of triadic closure, popularity bias, echo chambers and polarization measured using the metrics presented in subsection II F and evaluated using the methodology presented in subsection II E. In the results presented next, we fixed the parameters in the same way as Cinus et al. (2022): We generated networks with 400 nodes and $\sim$ 5500 edges, with a degree distribution following a power law with an exponent $\alpha = 2.75$ and average degree of $k = 12$ in accordance with what it has been observed in the literature for real social networks (Szüle et al. 2014). We set $R_{max} = 0.4 \cdot |E|$ in order to avoid disconnected graphs after running PROD, and we set $T_{max} = 5000$. The number of interactions per node is set to $S = 2$ to allow for mixed interactions (with and without recommended links). The internal parameters of the ODM are set to $\mu_c = \epsilon = 0.2$. For the PPR and WTF recommenders, we use a damping factor $\alpha_p = d = 0.85$, for the OBA recommender we set $\gamma = 1.6$. With the previous parameters, we perform the steps outlined in subsection II E running $K = 500$ simulations with each of the recommenders in order to quantify the effect of the recommenders over a sequence of networks with different initial values of modularity and homophily gien by $\mu \in \{0.05, 0.55, 0.95\}$ and $\eta \in \{0.2, 0.5, 0.8\}$.

In Figures 2 and 3 we present our models' results for all of the metrics and recommendation algorithms presented in subsections II F and II D, with the exception of the Node2Vec algorithm, since due to computational issues, we only present qualitative results for this case in Appendix B. In both of these figures, the x-axis corresponds to the initial homophily value η, and the y-axis corresponds to the initial modularity value μ of the networks taken into account. Therefore each region of the heatmap represents the value for the average difference Δm between a measure with and without a friend recommender. The color of the regions represents the magnitude of Δm, where regions with an intense red color represent an increase in measure m when using a recommendation algorithm, and regions with an intense blue color represent a decrease in measure m when using a recommendation algorithm. Each region of the heatmaps is marked with a number only if it is significant ($p < 0.001$) according to the Kolmogorov-Smirnoff Test.

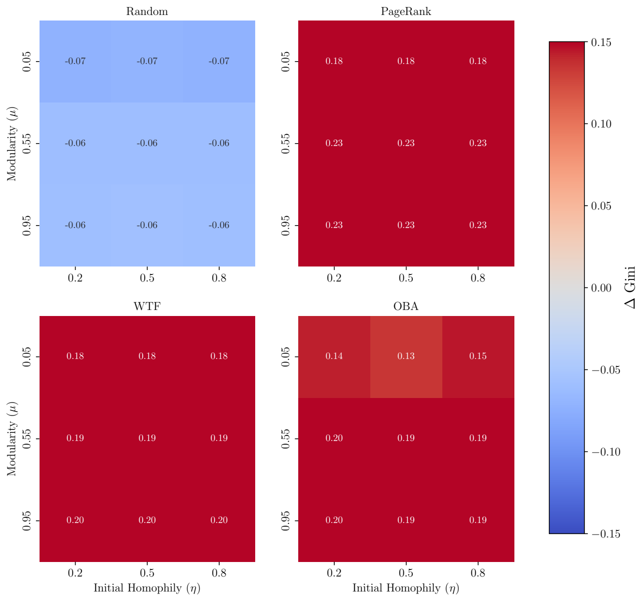
In the following subsections, we will present an discuss the results obtained for all recommender systems regarding triadic closure using the clustering coefficient, popularity bias using the Gini coefficient of the indegree distribution, the echo chamber effect using the NCI score, and polarization using the RWC score. As a complement, we will also propose some idealized intervention policies taken from Cinus et al. (2022) and we will assess their effectiveness in diminishing the effect of recommendation algorithms under the PROD framework.

A. Triadic Closure & Popularity Bias

In Figure 2a we present our models' results for the clustering coefficient. In this case, we can observe that



(a) ΔClustering.



(b) ΔGini.

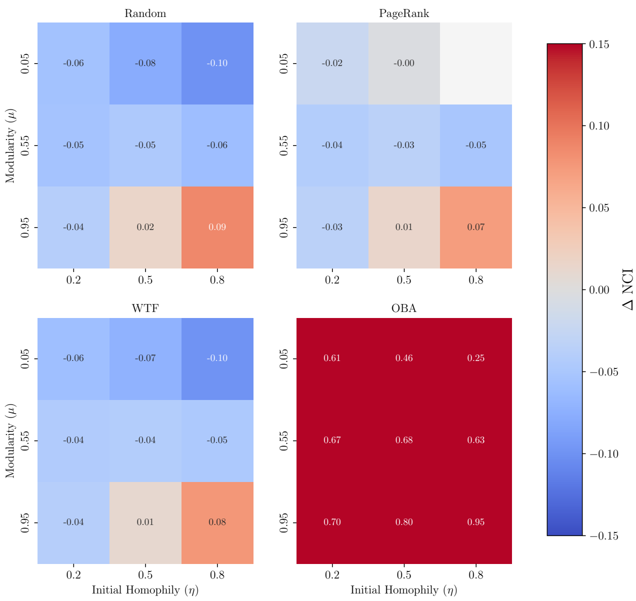
Figure 2: Simulation results for the clustering coefficient and the in-degree Gini coefficient of the evaluation method presented in subsection II E. Each heatmap represents a recommendation algorithm and colors represent the Δm value under different pairs of modularity μ and initial homophily η.

all recommendation algorithms, except for the random recommender, significantly increase the clustering coefficient for some initial conditions. In particular, we can observe that in all cases (except for the random case), we observe an increase in the clustering coefficient whenever $\mu \geq 0.55$, no matter what the initial homophily value η. This is an interesting result, since it means that for networks that are not already segmented in tightly interconnected clusters, the recommendation algorithms makes the networks more cohesive as more triangles are getting closed. In the case where networks have a highly modular structure, and therefore already have a high clustering coefficient, we observe that the effect of the recommendation algorithm is the opposite as before, and clustering coefficient decreases as a result of the algorithm since this algorithm tends to *reshuffle* the links in the already formed communities. It is important to also notice that the PPR algorithm is the algorithm that increases more significatively the clustering coefficient since by nature, this algorithm will tend to recommend nodes that are within the neighborhood of a given node and thus triangles are closed more frequently.

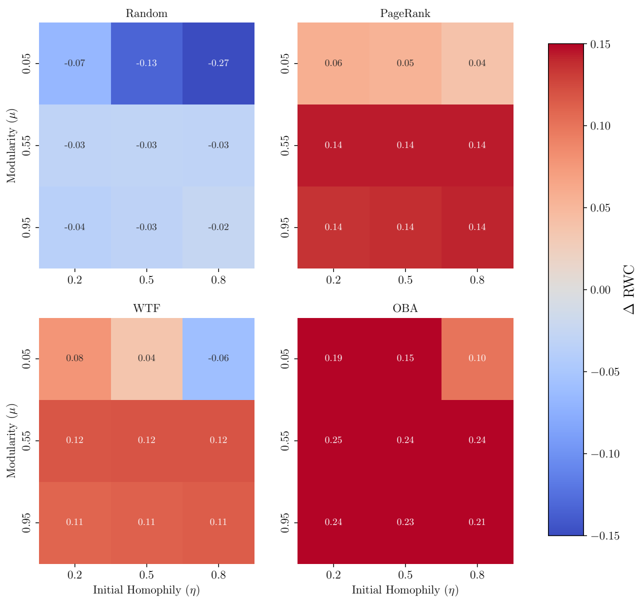
In Figure 2b we present our models' results for the Gini coefficient of the in-degree distribution. In this case, we observe that for all recommendation algorithms, except for the random recommender, there is a significant increase of the Gini coefficient for the in-degree distribution no matter the values of μ and ν, meaning that no matter the initial structure of the network, the recommendation algorithms presented in this project do reinforce the popularity bias effect by making popular nodes (high in-degree) more popular. This clearly doesn't happen when performing random recommendations, since nodes with high in-degree are in no way more important than nodes with low in-degree under this schema.

B. Echo Chambers & Polarization

In Figure 3a we present our models' results for the NCI score. In this case we can observe that all recommendation algorithms, including the random recommender, significantly increase the NCI score for initial homophily values of $\eta \geq 0.5$ and modularity values of $\mu \geq 0.95$. This is consistent with the previous results and with Cinus et al. (2022)'s results, since it means that for highly homophilic but not modular networks, the effect of friend recommendation algorithms is to increase the echo chamber effect with respect to the null case without recommenders. We can observe an opposite behavior for the random and WTF recommenders for all the other values of η and μ, and specially stronger when η is high and μ is low, i.e. for highly homophilic and highly modular networks, since in this case it is



(a) ΔNCI.



(b) ΔRWC.

Figure 3: Simulation results for the NCI score and the RWC score of the evaluation method presented in subsection II E. Each heatmap represents a recommendation algorithm and colors represent the Δm value under different pairs of modularity μ and initial homophily η.

expected that the initial network already has an echo-chamber-like structure and the recommendation algorithms can disrupt it by reshuffling links. We can also observe that the OBA recommender presents the greatest increase for the NCI score with respect to the no-recommendations case, this is expected since the purpose of this recommendation algorithm is to link nodes with similar opinions and can act as a benchmark algorithm in this context.

In Figure 3b we present our models' results for the RWC score. In this case we can observe that all recommendation algorithms, except the random recommender, significantly increase the RWC score for modularity values of $\mu \geq 0.55$ regardless of the initial homophily values η. This is consistent with previously found results and the results obtained by Cinus et al. (2022) since again, for networks that are not already segmented in tightly interconnected clusters, the recommendation algorithms increase polarization of opinions with respect to the case where no recommendation takes place. On the other hand, for networks segmented in clusters, we observe little-to-none effect of the recommendation algorithms with the exception of the random recommender, which has an opposite effect on polarization by performing random shuffling of edges. Finally, for the OBA recommender we observe a similar behavior as in the echo chamber case, since it clusters nodes with the same opinions together.

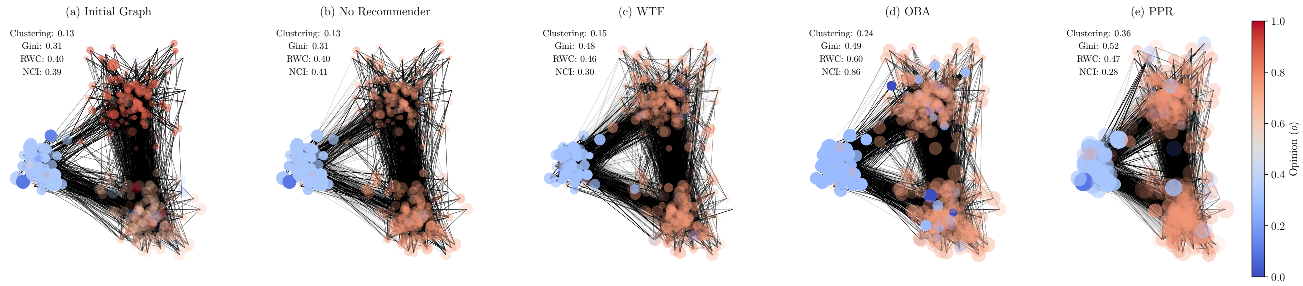
C. General Results

In general, we can observe that all the studied recommendation algorithms exacerbate the node popularity bias, as quantified by the Gini coefficient of the in-degree distribution, and have a significant effect on the closing of triangles, the formation of echo chambers and polarization of opinions whenever the following conditions are met:

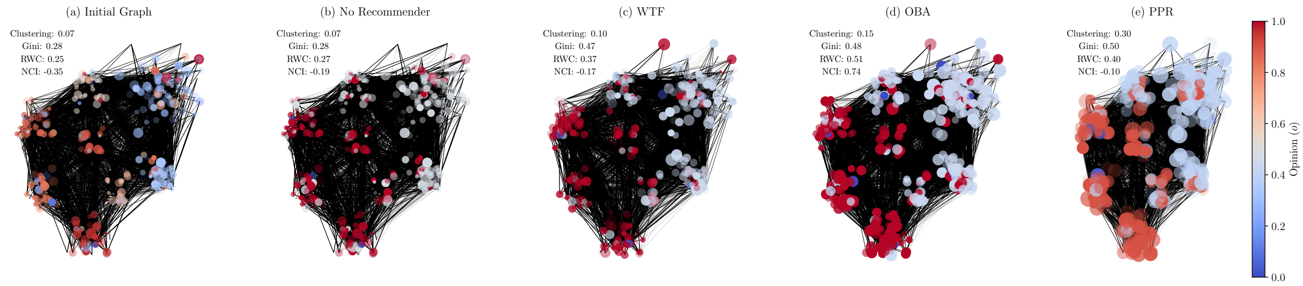
(1) The network has not already a highly segregated modular structure ($\mu \geq 0.95$).

(2) The network has highly homophilic links, i.e. nodes tend to connect with others with the same opinions ($\eta \geq 0.8$).

Moreover, we also observed that the effect of recommendation algorithms on highly modular networks with or without homophilic links is negligible in most of the cases (except for the OBA recommender) and it has even the opposite expected effect and it reduces the echo chamber effect and polarization on the network since the network in this cases is already highly polarized to begin with. In Figure 4 we present an example of the evolution of two networks of interest: a highly modular and homophilic network, and a non-modular and homophilic network. In this case, we observe qualitatively the results described above: in the the highly



(a) Highly polarized network with $\mu = 0.25$ and $\eta = 0.8$.



(b) High homophily, low modularity network with $\mu = 0.95$ and $\eta = 0.8$.

Figure 4: Evolution of networks with different initial conditions and different recommender systems. From left to right we observe the initial network (a) after: no recommendations (b), WTF recommendations (c), OBA recommendations (d), and PPR recommendations (e). Colors represent the opinion of each node, node size represents the clustering coefficient and the transparency represents the correlation of each node with its neighbors. Each community has a defined position on the plot to improve visibility. By observing both rows, we can see no significant difference on the evolution of the highly polarized network (above row) in comparison with the case of a highly homophilic but with low modularity case (below row).

modular and homophilic case, we have a negligible effect of recommendation algorithms regarding the formation of echo chambers and polarization. On the other side, for the non-modular case, this effect is more notorious as the network gets more segregated and nodes tend to cluster with similar nodes. This tendency is qualitatively similar also for the Node2Vec recommendation system (see Appendix B and Figure 7) suggesting a similar behavior for random-walk based recommendation algorithms.

D. Intervention Policies

In order to conclude this project, we also study how to use the PROD framework to evaluate intervention policies that could potentially mitigate the effects of recommendation algorithms on the formation of echo chambers and polarization, along with the exacerbation of popularity bias. We define the intervention procedure to work on top of the modified-PROD algorithm (see algorithm 1) whenever there is a recommendation. In particular, given a node u and a recommended node v given by a recommendation algorithm l, we define an intervention probability ξ such that node v is replaced by another one selected according to a probability distribution P_v. This probability distribution represents the intervention policy. In this project we explore two idealized intervention policies as defined by Cinus et al. (2022):

(1) *Random uniform policy:* In this case, the probability distribution P_v is the uniform distribution over the non-neighbor set of nodes. Therefore, a recommendation is modified by selecting another node at random.

(2) *Opinion diversity policy:* In this case, the probability distribution P_v is defined such that the probability of selecting a node is proportional to the difference between the target node u and the potential node v: $p_v = |o_v - o_u|$. In this case, the intervention policy favors recommendations with different opinions.

Although this recommendation policies could have little applicability in the real world, since users tend to

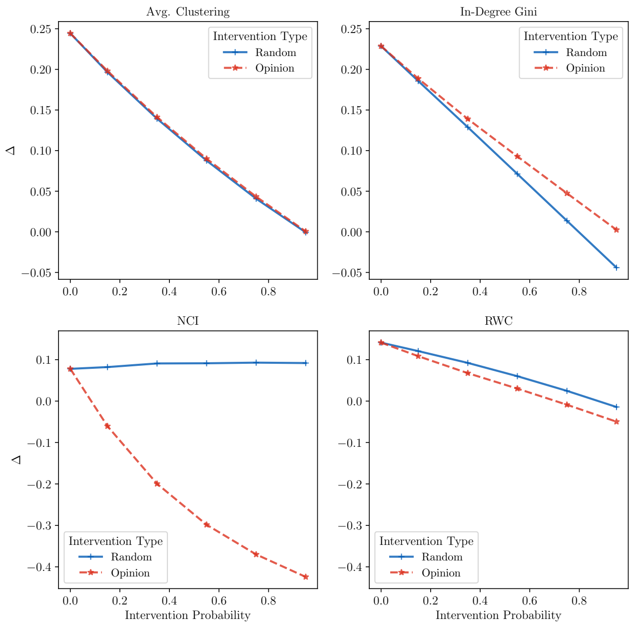


Figure 5: Δm for the clustering coefficient, Gini coefficient of the in-degree distribution, NCI, and RWC scores given random intervention and opinion diversity intervention with different probabilities. The results presented in this figure correspond to a random network with modularity $\mu = 0.95$ and homophily $\eta = 0.8$ (low-modular and homophilic network).

accept recommendations of well-connected nodes along with nodes with the same opinions, this exploration highlights the methodology one could follow to test more complicated intervention policies in order to try to mitigate some effect emerging from the recommendation algorithms acting on a social network. For simplicity, we only test these intervention strategies on the PPR recommendation algorithm. In Figure 5 we observe the results of the intervention policies for all the measures studied before on a low-modular and highly homophilic network receiving recommendations from a PPR recommendation algorithm.

As we can observe, the random intervention and the opinion-diversity intervention are equally efficient in reducing the difference of the clustering coefficient since this metric doesn't depend on the opinion of nodes and therefore both strategies perform an effective pseudo-random rewiring of the network. For the Gini coefficient of the in-degree distribution, we observe a similar behavior, although the random intervention is slightly better at reducing the difference of this quantity for higher intervention probabilities. For the case of the NCI, we observe a that the opinion diversity intervention is very effective in prominently reducing the effect of the PPR recommendation algorithm on the formation of echo chambers since it is purposely defined to reduce the formation of links of nodes with similar opinions. In this case, we can observe that even low values of intervention probability ξ are enough to significantly reduce this effect. On the other side, we can observe that the random strategy doesn't reduce the effect of the recommendation algorithm and instead the Δ is mantained fairly constant even for high intervention probabilities. This is a surprising result, since even though it was expected to be a worse strategy than the opinion-diversity strategy, it was expected also to have a decrease in some sense. Finally, for the case of polarization, quantified by the RWC, we observe a very slow reduction of the effect of the recommendation algorithm for both strategies, although a slightly better performance for the opinion diversity strategy, as expected.

IV. CONCLUSIONS

In this project, we reproduce and extend the results obtained by Cinus et al. (2022) using a modified version of the PROD framework to assess the effect of friend recommender systems on popularity bias and the formation of echo chambers and polarization on social networks modeled by a real-world random network model and an ODM. We study different scenarios by varying the modularity and homophiliy of these networks and observe that popularity bias is exacerbated for all studied recommendation algorithms (except the random case) and the emergence of echo chambers and polarization is reinforced for networks with low modular structure, but highly homophilic links, whereas in the case of highly modular structure, the effect of recommenders is negligible and sometimes it is beneficial for the reduction of the echo chamber effect.

One limitation of this project is the lack of generalization, since in this case we didn't assess different ODM or network structures. Nevertheless, we Cinus et al. (2022) tested the PROD methodology for different cases, finding similar results in all of them. Moreover, we did modify the PROD algorithm with respect to the original implementation, observing that even when making different assumptions, the main results remain qualitatively the same. We also didn't contrast our results with real-world data, as this is very difficult in practice and remains one of the most important challenges towards building fair and socially responsible algorithms.

We conclude that even though is very computationally demanding, the PROD methodology can be effectively used to model and measure the effect of recommendation algorithms when interacting with social networks, and even to propose and test the effectiveness of intervention policies that aim to reduce such an effect as a sort of playground model before testing on real data. It remains as future work to test more complicated algorithms like network embedding techniques and the quantitative assessment of the Node2Vec algorithm.

H. Abdollahpouri. Popularity bias in ranking and recommendation. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '19, page 529–530, New York, NY, USA, 2019. Association for Computing Machinery. doi:10.1145/3306618.3314309.

C. Bagchi, F. Menczer, J. Lundquist, M. Tarafdar, A. Paik, and P. Grabowicz. Social media algorithms can curb misinformation, but do they? 2024. doi: 10.5281/ZENODO.13787981.

A.-L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999. doi: 10.1126/science.286.5439.509.

M. Cinelli, G. D. F. Morales, A. Galeazzi, W. Quattrociocchi, and M. Starnini. The echo chamber effect on social media. *Proceedings of the National Academy of Sciences*, 118(9):e2023301118, 2021. doi:10.1073/pnas.2023301118.

F. Cinus, M. Minici, C. Monti, and F. Bonchi. The effect of people recommenders on echo chambers and polarization. *Proceedings of the International AAAI Conference on Web and Social Media*, 16:90–101, 05 2022. doi: 10.1609/icwsm.v16i1.19275.

P. Dandekar, A. Goel, and D. T. Lee. Biased assimilation, homophily, and the dynamics of polarization. *Proceedings of the National Academy of Sciences*, 110(15):5791–5796, 2013. doi:10.1073/pnas.1217220110.

G. Deffuant, D. Neau, F. Amblard, and G. Weisbuch. Mixing beliefs among interacting agents. *Advances in Complex Systems*, 3:87–98, 01 2000. doi:10.1142/S0219525900000078.

R. Dunbar. Neocortex size as a constraint on group size in primates. *Journal of Human Evolution*, 22(6):469–493, 1992. doi:https://doi.org/10.1016/0047-2484(92)90081-J.

L. Espín-Noboa, C. Wagner, M. Strohmaier, and F. Karimi. Inequality and inequity in network-based ranking and recommendation algorithms. *Scientific Reports*, 12(1), Feb. 2022. doi:10.1038/s41598-022-05434-1.

F. Fabbri, M. L. Croci, F. Bonchi, and C. Castillo. Exposure inequality in people recommender systems: The long-term effects, 2021.

G. Fagiolo. Clustering in complex directed networks. *Physical Review E*, 76(2), Aug. 2007. ISSN 1550-2376. doi: 10.1103/physreve.76.026107.

A. Ferrara, L. Espin-Noboa, F. Karimi, and C. Wagner. Link recommendations: Their impact on network structure and minorities. In *14th ACM Web Science Conference 2022*, WebSci '22, page 228–238. ACM, June 2022. doi: 10.1145/3501247.3531583.

K. Garimella, G. D. F. Morales, A. Gionis, and M. Mathioudakis. Quantifying controversy on social media. *Trans. Soc. Comput.*, 1(1), Jan. 2018. doi:10.1145/3140565.

C. Gini. *Variabilità e mutabilità: contributo allo studio delle distribuzioni e delle relazioni statistiche. [Fasc. I.]*. Studi economico-giuridici pubblicati per cura della facoltà di Giurisprudenza della R. Università di Cagliari. Tipogr. di P. Cuppini, 1912.

D. F. Gleich. Pagerank beyond the web, 2014.

M. S. Granovetter. The strength of weak ties. *American Journal of Sociology*, 78(6):1360–1380, 1973.

A. Grover and J. Leskovec. node2vec: Scalable feature learning for networks, 2016.

P. Gupta, A. Goel, J. Lin, A. Sharma, D. Wang, and R. Zadeh. Wtf: the who to follow service at twitter. In *Proceedings of the 22nd International Conference on World Wide Web*, WWW '13, page 505–514, New York, NY, USA, 2013. Association for Computing Machinery. ISBN 9781450320351. doi:10.1145/2488388.2488433.

I. Guy. *People Recommendation on Social Media*, pages 570–623. 05 2018. ISBN 978-3-319-90091-9. doi:10.1007/978-3-319-90092-6_15.

A. Kumar, S. Singh, K. Singh, and B. Biswas. Link prediction techniques, applications, and performance: A survey. *Physica A: Statistical Mechanics and its Applications*, 553: 124289, 02 2020. doi:10.1016/j.physa.2020.124289.

A. Lancichinetti, S. Fortunato, and F. Radicchi. Benchmark graphs for testing community detection algorithms. *Physical Review E*, 78(4), Oct. 2008. ISSN 1550-2376. doi: 10.1103/physreve.78.046110. URL `http://dx.doi.org/10.1103/PhysRevE.78.046110`.

R. Lempel and S. Moran. Salsa: the stochastic approach for link-structure analysis. *ACM Trans. Inf. Syst.*, 19(2):131–160, Apr. 2001. ISSN 1046-8188. doi: 10.1145/382979.383041.

S. Lewandowsky, U. K. Ecker, and J. Cook. Beyond misinformation: Understanding and coping with the "post-truth" era. *Journal of Applied Research in Memory and Cognition*, 6(4):353–369, 2017. ISSN 2211-3681. doi: https://doi.org/10.1016/j.jarmac.2017.07.008.

M. McPherson, L. Smith-Lovin, and J. M. Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27:415–444, 2001.

T. Mikolov, K. Chen, G. S. Corrado, and J. Dean. Efficient estimation of word representations in vector space. In *International Conference on Learning Representations*, 2013. URL `https://api.semanticscholar.org/CorpusID:5959482`.

L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking : Bringing order to the web. In *The Web Conference*, 1999. URL `https://api.semanticscholar.org/CorpusID:1508503`.

W. Quattrociocchi, A. Scala, and C. Sunstein. Echo chambers on facebook. *SSRN Electronic Journal*, 01 2016. doi: 10.2139/ssrn.2795110.

J. Szüle, D. Kondor, L. Dobos, I. Csabai, and G. Vattay. Lost in the city: Revisiting milgram's experiment in the age of social networks. *PloS one*, 9:e111973, 11 2014. doi: 10.1371/journal.pone.0111973.

A. Sîrbu, D. Pedreschi, F. Giannotti, and J. Kertész. Algorithmic bias amplifies opinion fragmentation and polarization: A bounded confidence model. *PLOS ONE*, 14(3):1–20, 03 2019. doi:10.1371/journal.pone.0213246.

Appendix A: LFR-Benchmark Model

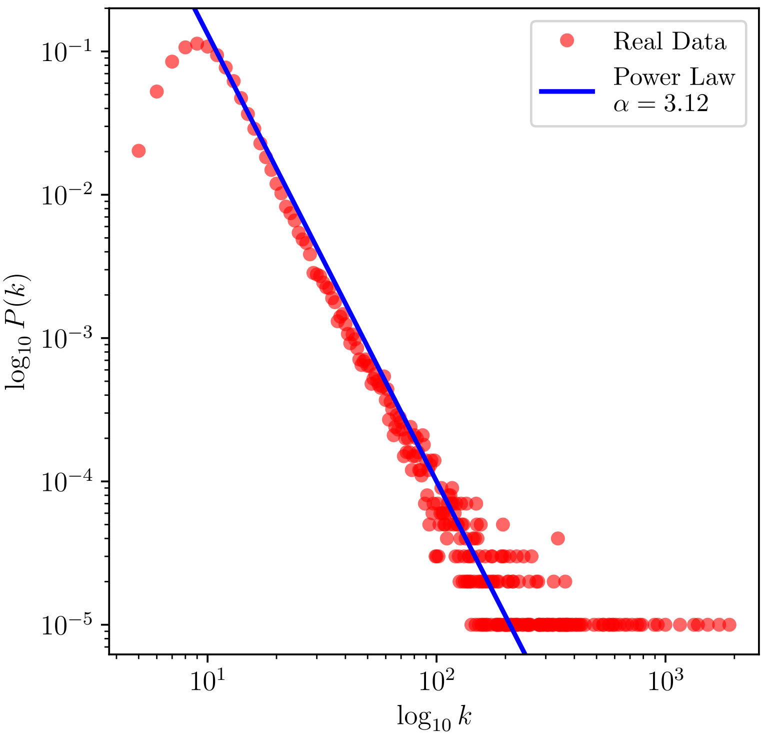


Figure 6: Degree distribution of a toy example of the original LFR-Benchmark model proposed by Lancichinetti et al. (2008) (in log-log scale) for $N = 10^5$. Theoretically, the model is generated to have a power-law degree distribution with coefficient $\alpha = 2.75$ and a high modular structure with $\mu = 0.1$ and a power-law community size distribution with $\beta = 1.1$. As we can observe, the model indeed presents a power-law degree distribution although the coefficient varies given the stochastic nature of the generative model.

Appendix B: Node2Vec Qualitative Results

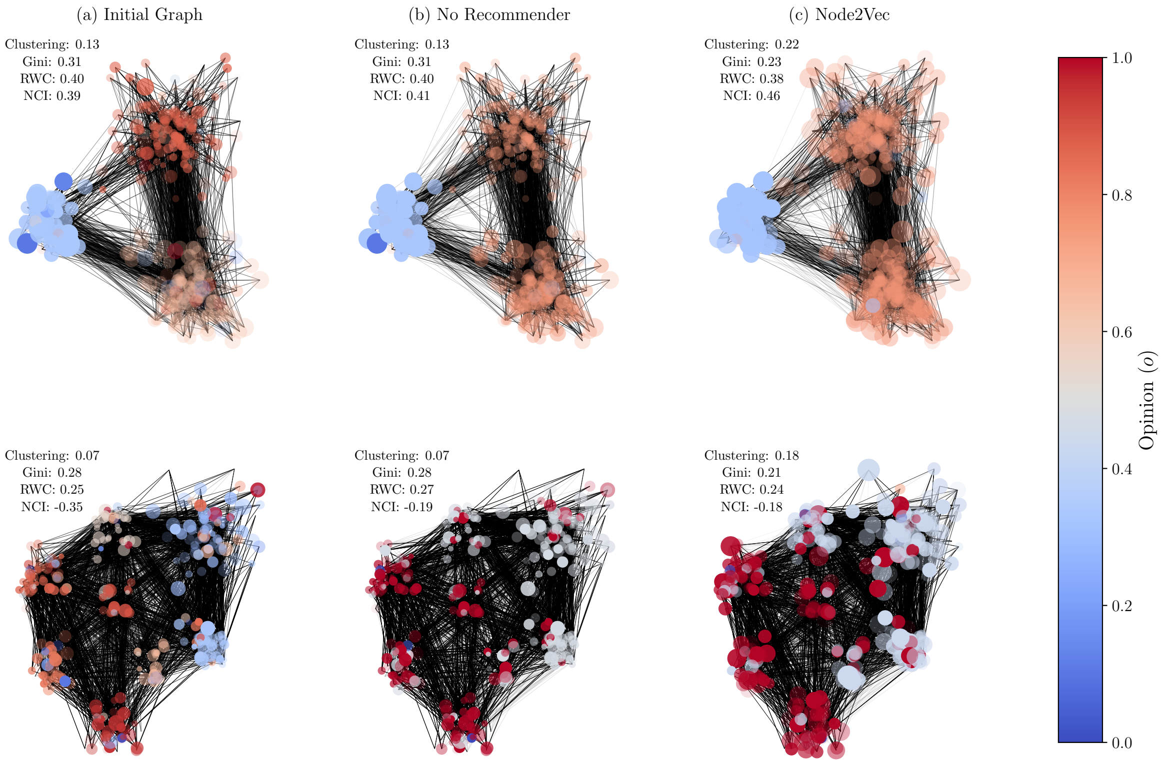


Figure 7: Evolution of networks with different initial conditions and different recommender systems. From left to right we observe the initial network (a) after: no recommendations (b), and Node2Vec recommendations (c). Colors represent the opinion of each node, node size represents the clustering coefficient and the transparency represents the correlation of each node with its neighbors. Each community has a defined position on the plot to improve visibility. By observing both rows, we can see no significant difference on the evolution of the highly polarized network with $\mu = 0.25$ and $\eta = 0.8$ (above row) in comparison with the case of a highly homophilic but with low modularity case with $\mu = 0.95$ and $\eta = 0.8$ (below row).