Assignment 2 – Gossip spreading in Reddit's Hyperlinks network

Miggiano Davide 4840761
Morando Andrea 4604844

Introduction

To carry out the second assignment, it was decided to start from the same graph of the first assignment, performing the exact same steps to reduce it to a number of nodes and arcs that would be interesting but less expensive from a computational point of view.

Unlike the first assignment after taking the largest SCC and having reduced the graph by taking the 40% of the best nodes given by the PageRank, it was decided to take one of the communities with a reasonable number of nodes and arcs. The best communities found are shown in the Table 1:

Community	Nodes	Edges	Average Degree
0	1667	11899	20.9124
1	1403	21698	40.7904
2	781	4019	17.8617

Table 1: Top 3 community

Given these options, it was decided to choose the first community as it has a good number of nodes and arcs and a relatively low average degree compared to the others, in order to have an easier spreading of the gossip.

This community's degree distribution is the one in Figure 1:

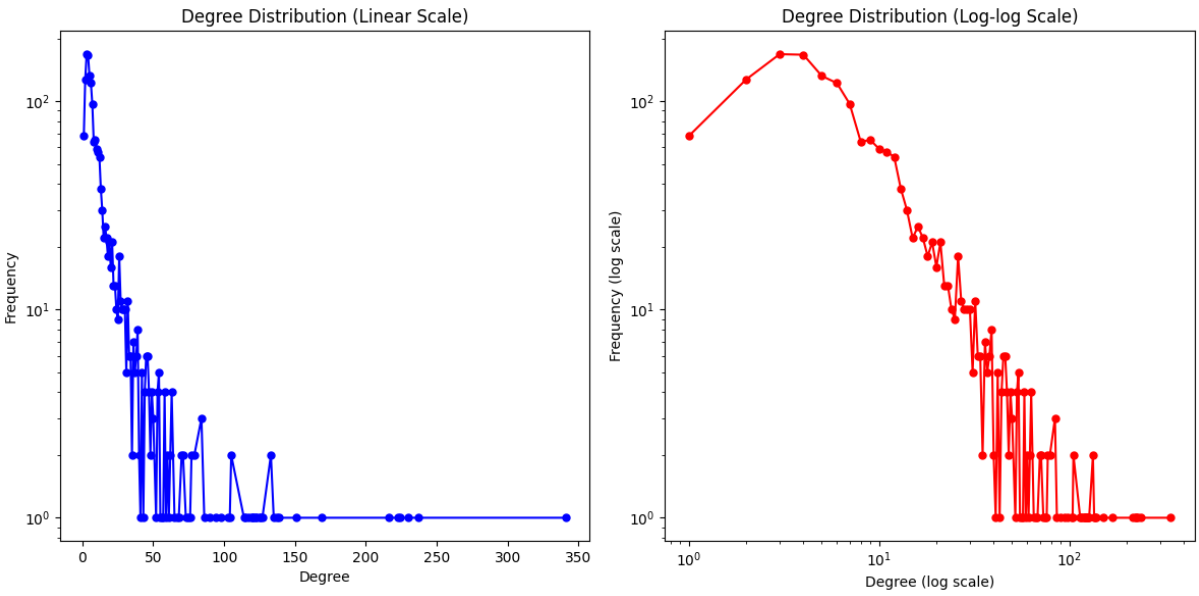

Figure 1: Selected community's degree distribution

Since our community has a degree distribution fairly similar to a power law, we decided to carry out several experiments: the first quite random and the others instead targeted according to the characteristics that we imagine to be most effective for contagion based on the first result. The function that calculate the contagion spreading is a synchronous implementation of the threshold model, in order to have a more controlled spreading over each steps without influencing the result with contagion's update during the calculation.

In all our experiments the contagion's threshold θ is fixed, and on our own initiative, we decided to have two different parameters, a first one related to "basic" nodes with θ=0.5 while for malicious nodes the parameter is θ=0.3. In our scenarios a malicious node will tend to disseminate information faster than a normal node (because of its interest in sabotaging the network). These parameters remain fixed for all experiments, because in our case the average degree was quite high (≈ 21), it was not necessary to carry out cases with higher (or lower) thresholds as the contagion might not occur or occur too quickly.

For example, if we use a $\theta = 0.4$ we reach a full spreading most of the times, with an amount of steps $\approx 25/30$ like in Figure 2:

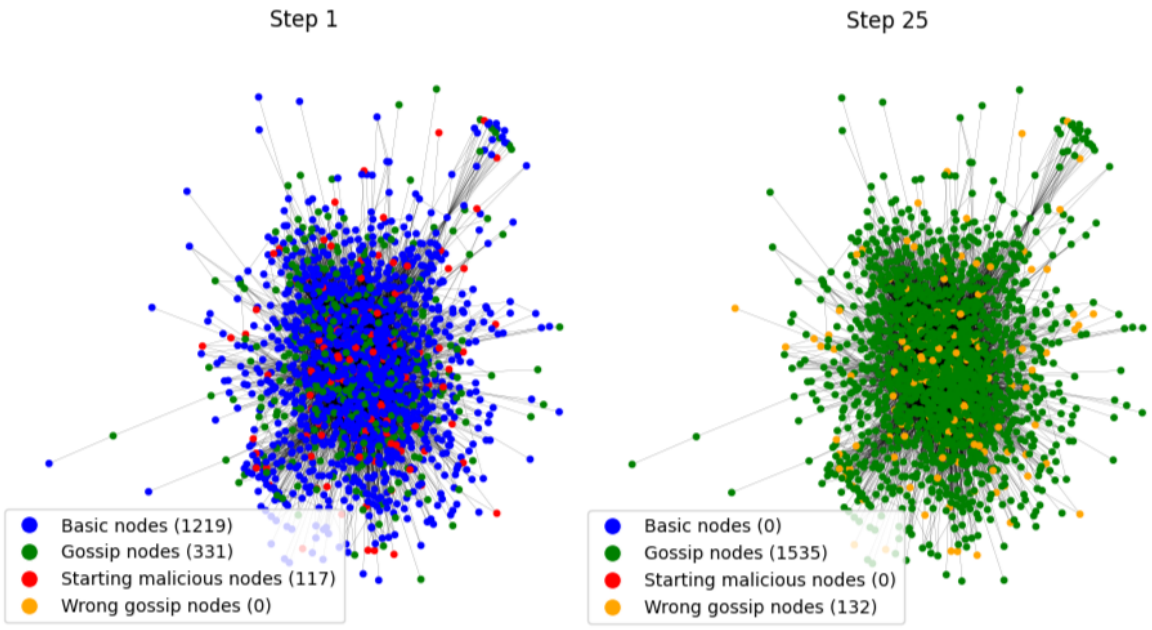

Figure 2: Selected community's degree distribution

As we can see for a better visualization of each contagious' step we used different color for the nodes:

- blue = node that have not yet received the gossip
- green = gossip nodes
- red = initial malicious nodes
- orange = nodes that receive or have spread the wrong gossip (the red nodes became orange when receive the gossip)

Random gossip node with different threshold and malicious node

For the first two cases, it was decided to use random nodes for both gossip and malicious node initialization, in each cases we modify only the quantity of malicious and gossip nodes playing with their probability.

In the first case the starting configuration provides a threshold for becoming a gossip node of 20% and a 7% of being a malicious nodes with respect to the total. In this case we have the following condition in Figure 3:

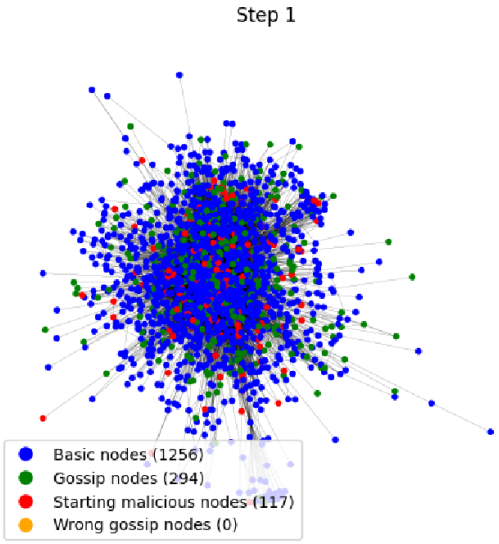

Figure 3: First case, first step

Since our network has a very high degree and the initial gossip nodes may not be hubs or important nodes, they cannot influence a sufficient portion of nodes, so this kind of configuration on average does not reach full contagion.

The first experiment final result (Figure 4) is:

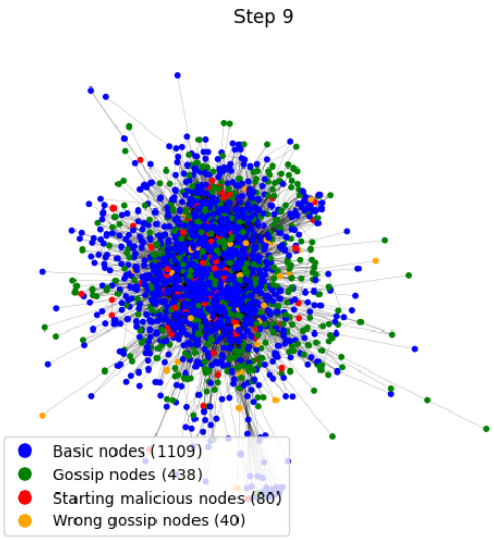

Figure 4: First case, last step

As can be seen, out of a total of 1256 basic nodes, we arrived at only 120 infections (of which only 3 received the wrong gossip)

In the second case, the threshold for becoming a gossip node is 30% while the number of malicious node is 10% of the total number of nodes. In this case the initial configuration is the one in Figure 5:

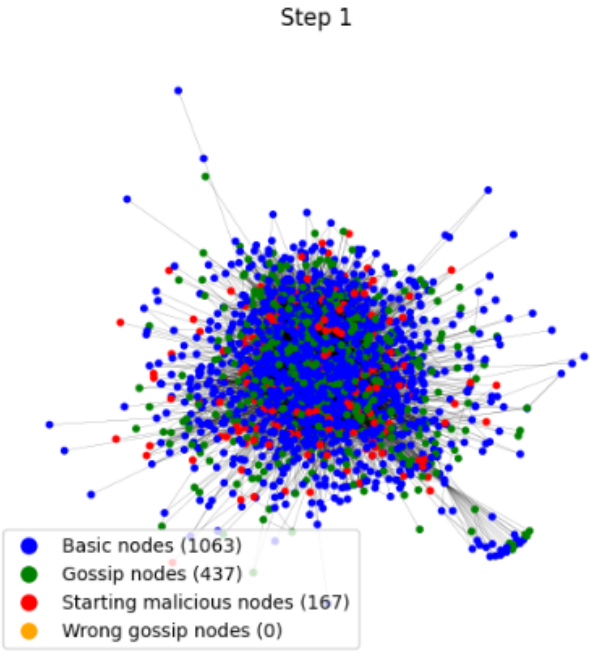

Figure 5: Second case, first step

The gossip nodes have increased just over a hundred, but unlike before, the contagion reaches a complete cascade in a total of 14 steps. Up to step 5 the contagions step by step increased by only 50 nodes, since then the contagions doubled to almost 100 nodes per step until the totality of the nodes is reached. Probably when enough gossip nodes manage to infect some important nodes like hubs (or nodes with higher betweenness) the information manages to pass to more groups in an accelerated manner. In Figure 6 we can have a graphical example:

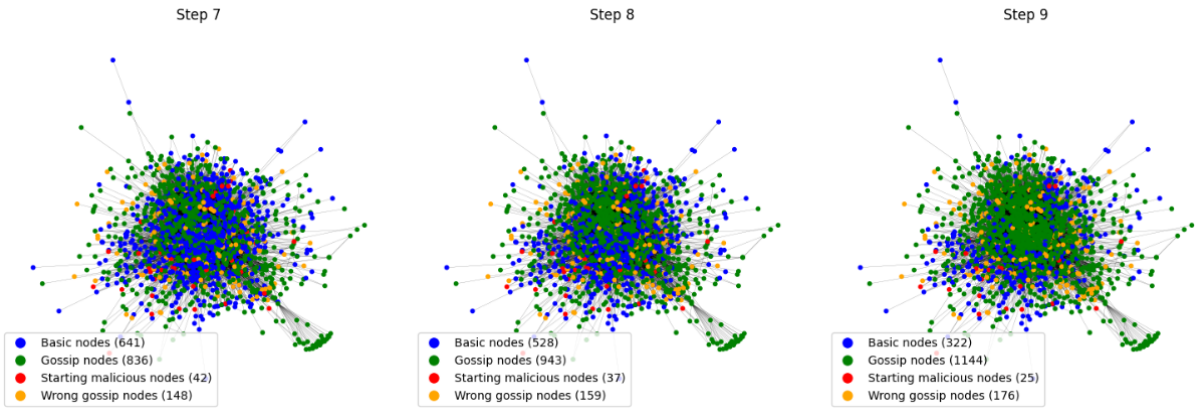

Figure 6: Faster contagious example

In the final configuration we have reached a total of 176 nodes with incorrect information, and 25 inactive malicious nodes, so from 167 initial malicious it translates into only 34 maliciously influenced. From this it follows that the malicious nodes, although in a non indifferent number, are unable, if randomly initialized, to disseminate the malicious information in an optimal manner

Hub selection as gossip or malicious node

In this case, we kept the amount of malicious nodes (still 10% of the total) but we tried to initialize the gossip nodes using the 100 nodes with the highest number of degree.

Already in the first 3 steps, it can be seen that this configuration is almost perfect for the rapid spread of contagion

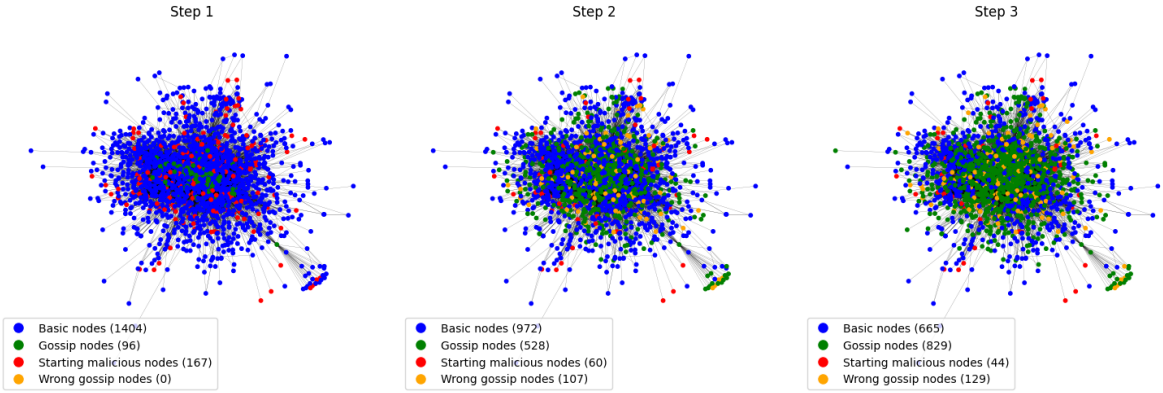

Figure 7: Hub gossip spreading

In just a few steps the model reaches around 8 times the amount of contagion compared to the starting gossip nodes, underlining how in our model the hubs play a key role. In this configuration, the contagion does not reach a complete cascade (about 10 nodes left) but it is so fast that malicious nodes can hardly influence the graph (in the final step the actual "wrong" contagions are about 6).

Since the role of the hubs is so decisive it was decided to reverse the roles, initializing the gossip nodes at random (with a threshold of 30%) but using as malicious nodes only 50 of the nodes with the highest degree. With this configuration, the contagion is in line with the previously analyzed cases; unlike before the malicious nodes remain "frozen" for many steps since to influence a hub it is required at least 30% of its neighbors to be gossip nodes. Up to step 6, the situation was as in Figure 8:

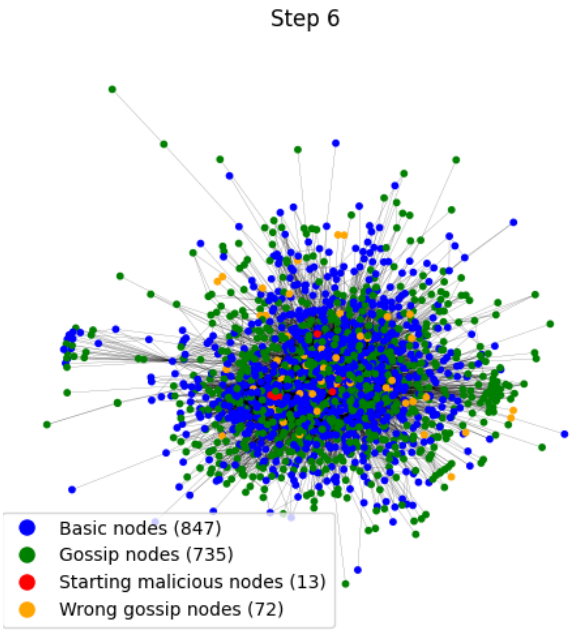

Figure 8: Hub malicious spreading, slow start

From the next step the gossip hubs start becoming more and more present, the contagion explodes and the hubs begin to influence all the remaining nodes, arriving in just 3 steps at the following situation shown in Figure 9:

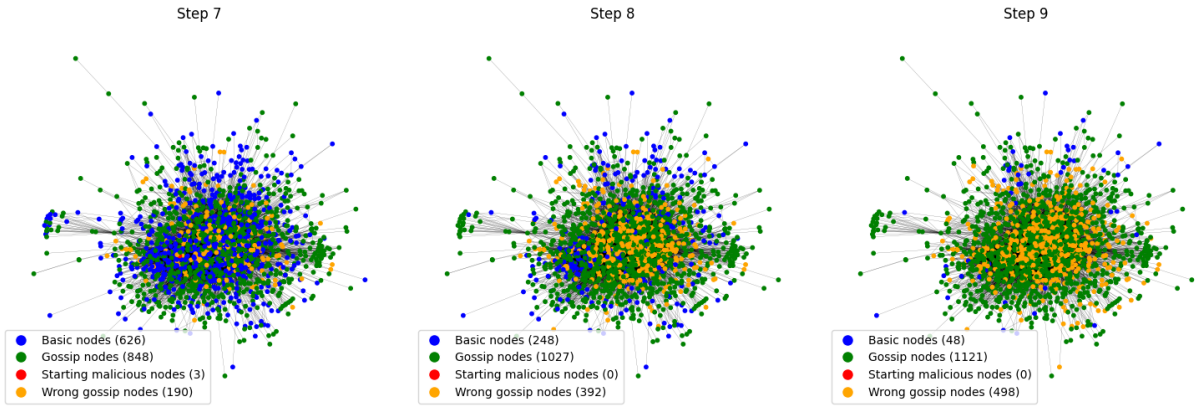

Figure 9: Malicious gossip contagion, hub explosion

The hub strategy proved to be very effective for passing malicious information because, as we can see, the total number of malicious contagion nodes is 498, reached within a complete cascade at step 12; but still almost 2/3 of the network received the right information. This obviously has to do with the fact that to influence a hub you need many gossip nodes among the neighbours and that therefore the malicious contagion is "delayed" compared to the basic one. With this in mind we decided to carry out one last experiment with the aim of increasing malicious contagions significantly while still maintaining an acceptable parameters.

Random gossip node and highest betweenness node as malicious

In this last case, we decided to use the 10% of nodes with the highest betweenness in order to try to spread the wrong gossip by using the same quantity of malicious nodes like in all other cases, but choosing the most effective ones for passing the contagion between the various areas of the network.

The initial configuration is as in Figure 10:

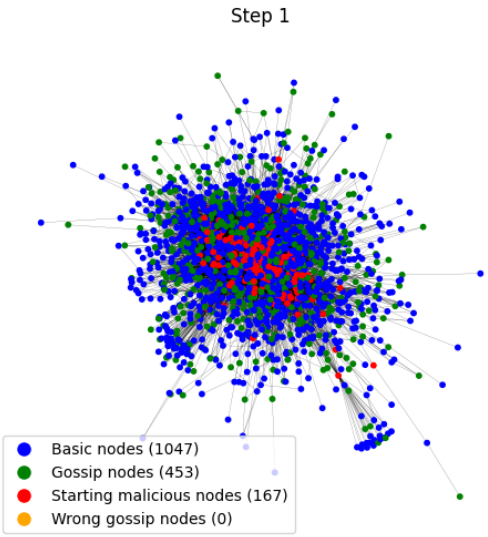

Figure 10: Higest betweenness as malicious node, starting configuration

In the first six steps, the infection continued without any particular difference from the initial cases; from step 9, however, specific perimeter areas began to become infected, with an increase in "wrong" infected nodes, which is very impactful compared to the malicious nodes not yet "activated"

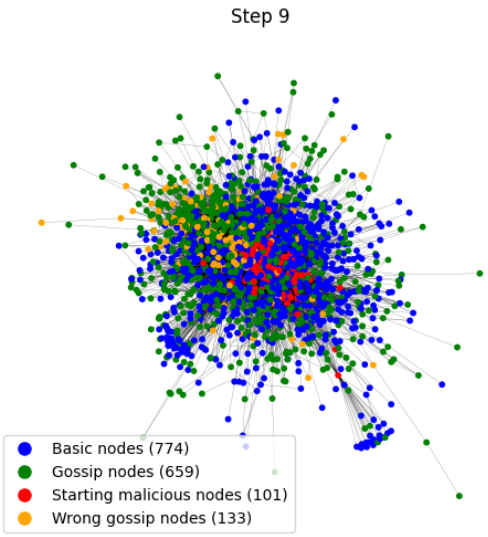

Figure 11: Higest betweenness as malicious node, starting configuration

In the next three steps, the malicious contagion explodes and reaches near convergence

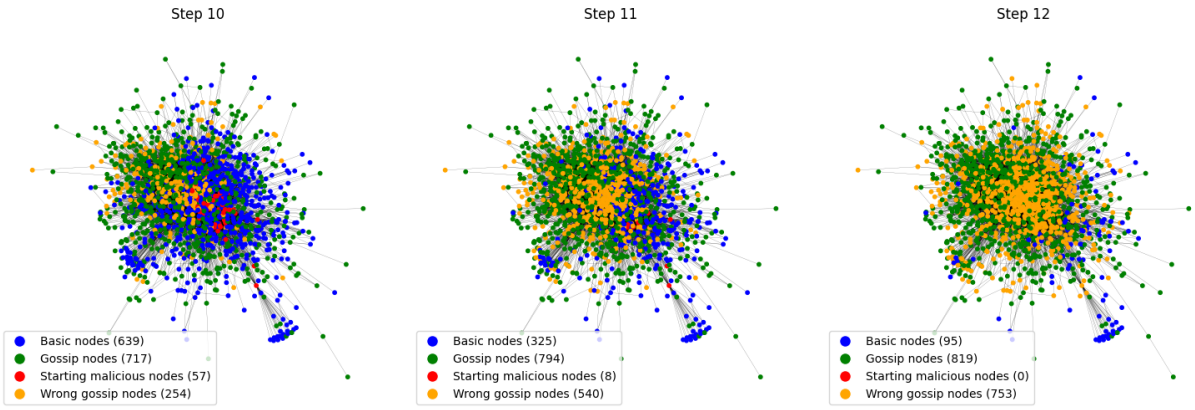

Figure 12: Malicious gossip fast spreading

as we can see in Figure 12, the malicious contagions double at each step until they almost reach the number of infected gossip nodes.

The contagion will reach the full cascade in just 2 steps with a total of malicious nodes exceeding the gossip one by approximately 20 nodes.

This configuration showed how the influence of the right nodes, with the same quantity, can lead from a contagion of about 35 wrong to one of 674 in not too many steps. This obviously also depends very much on the structure of the network; in our case Reddit is a social network strongly divided into communities, so we expected that the influence of hubs or nodes with good betweenness would be crucial.

Conclusion

Using a community of subreddit highlights the impact of node characteristics on contagion dynamics. As we can see during the experiments the random initialization did not lead to a good result because the average degree was too high and the structure of the network did not help in the gossip spreading. Strategies that target high-degree or high-betweenness nodes can significantly alter the spread of information, offering insights into both promoting beneficial information and mitigating the effects of malicious spread.