Abstract

The purpose of this report is to investigate the difference in effects of the likelihood of a user to like a post and the likelihood of a user to follow the original user. I believed that when treating each independently, a higher likelihood to like the post will have a more significant effect on the number of likes after multiple posts. As it turns out, this is true to start with, but after a point this loses all significance, as the effect of followers is lost at high like probabilities.

Background

In a social network, the number of likes a post gets is heavily dependent on the number of people that see it. In the simplified simulation of a social network that I have implemented, there are two factors that affect how many people see a post. Firstly, the number of people that follow the original poster, and secondly, the number of people that ‘like’ the post after seeing it.

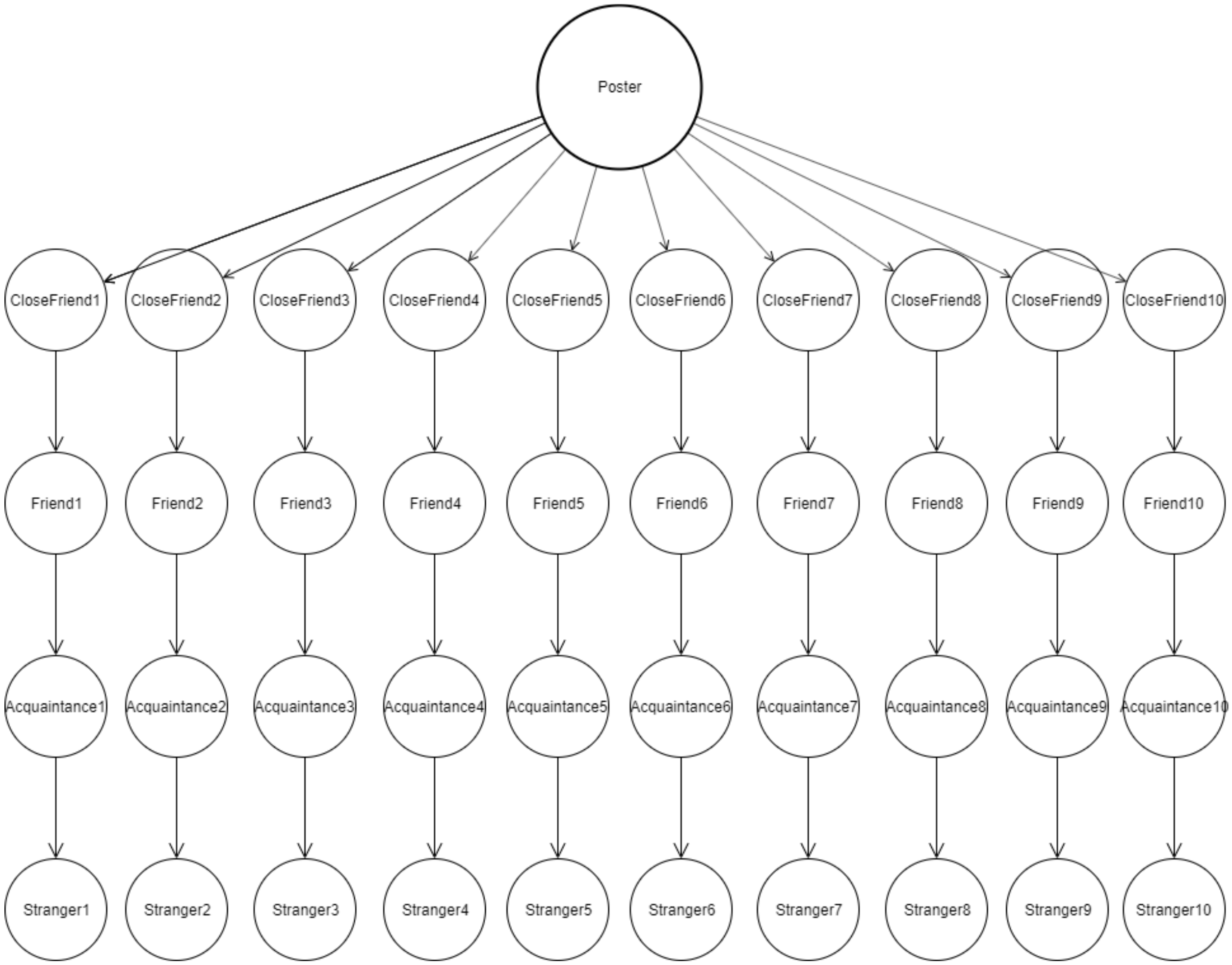
The simulations takes inputs for the percentage chance for each of these events, and determines per post how many likes it gets. I believe that increasing the chance of a like has a more significant domino effect on the final number of likes than if the probability to follow was increased. This effect will be most noticeable after multiple posts.

My reasoning behind this is that a user liking a post instantly connects all of their followers to the post. Also, equal probabilities for following and liking would result in more likes, as a follow is only possible if a like has already occurred.

Methodology

I plan to have two independent tests, one where I alter the probability to like, and one where I alter the probability to follow. When testing the follow probability, I will keep the like probability at 50%, and vice versa for testing like probability.

A diagram of the network is shown below, I created my own network so as to eliminate any unwanted variables such as clickbait or hard-coded follow events. As a post propagates it will move down layers away from Poster, as more posts are made, more follows will be created and the post will propagate further down. The poster will make six posts, one after another, and the number of likes for the first post will be compared to the number of likes of the sixth. This difference in likes is what we will compare for each different setting for like/follow probability. The number of posts being six was chosen as any more would max out the layers of the network, and any less would be less reliable data as the difference between the first and last posts will be smaller.



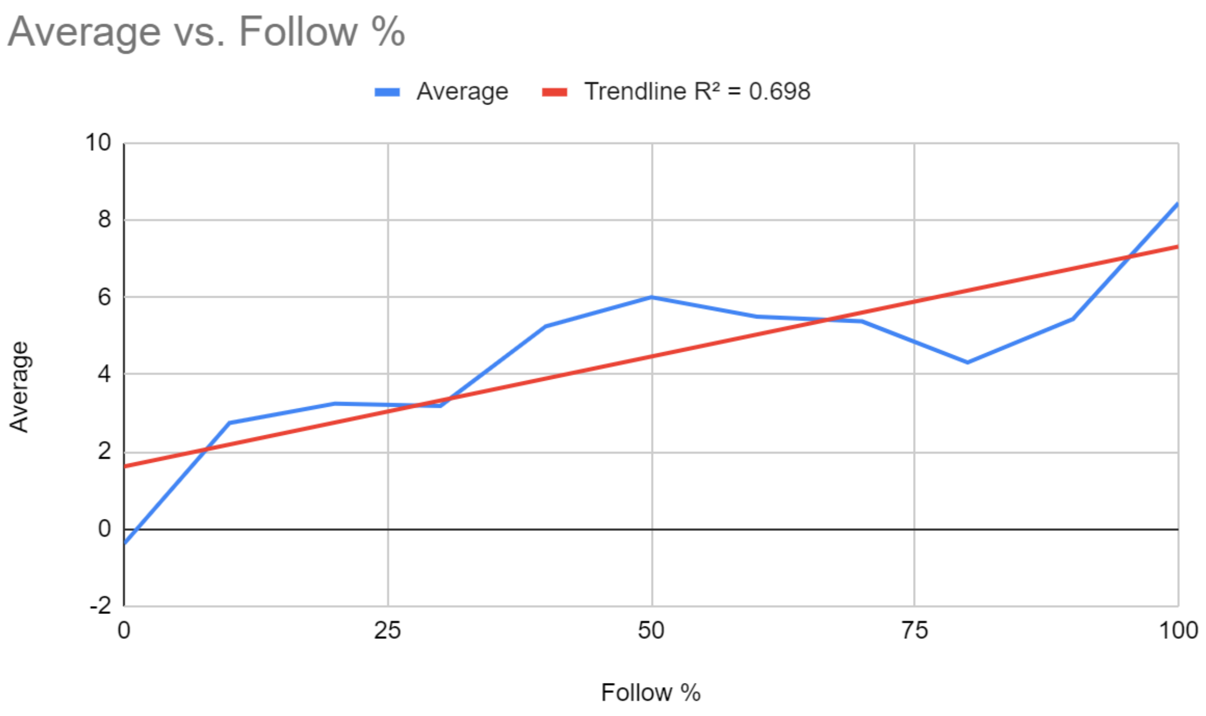
Because this experiment is based on two random variables, I will be taking many trials and comparing the averages for each probability.

To run this experiment I will be using SocialSim in simulation mode, as this will automatically step through the program and run every event. To make data collection easier, I have slightly modified by program to output to the terminal the difference in likes between the 1st and 6th post once the simulation finishes. The network and event files are included in this submission with the names netTest.txt and eventTest.txt. For example, to run the test with like probability as 50% and follow probability at 80%, I would use `java SocialSim -s netTest.txt eventsTest.txt 0.5 0.5`. The program as a whole has a relatively large memory overhead due to the many linked lists used for data storage and post propagation, although the entire program runs automatically in less than a second in simulation mode so it is not usually a problem. In terms of time complexity, the log file is being appended to multiple times per timestep, this does end up slowing down the program but not in any noticeable way for most datasets.

Results

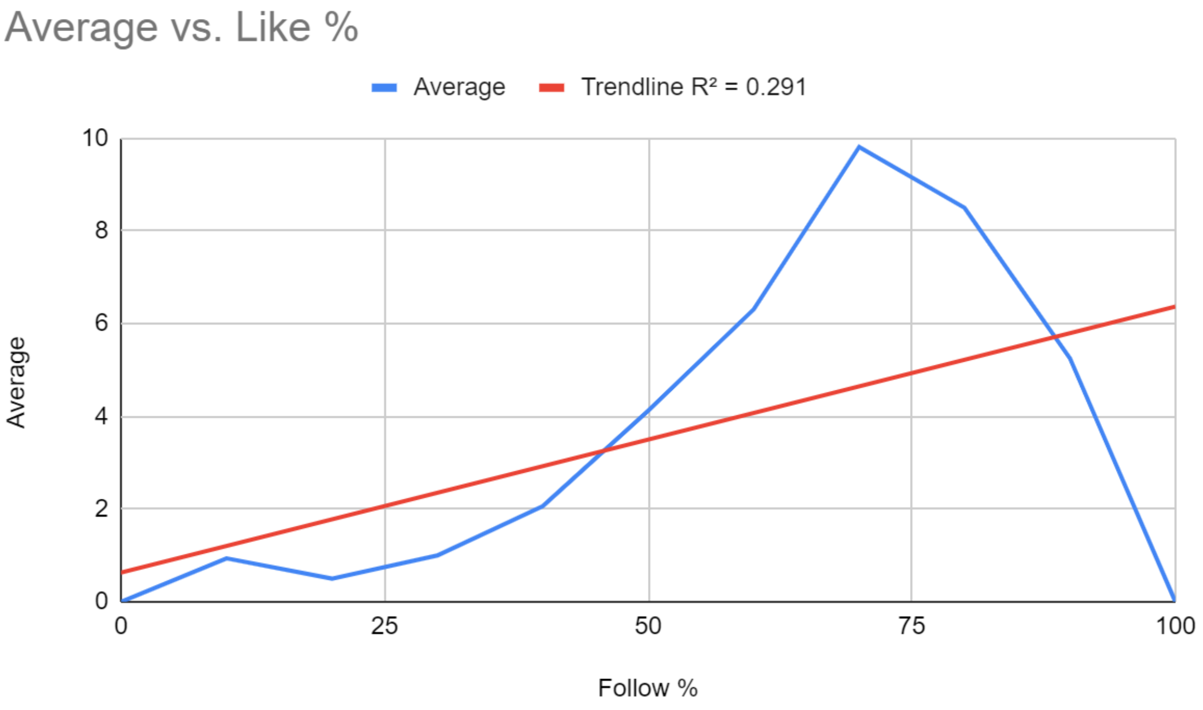
Variable Follow Probability, Like Probability 50%

	Trials. Difference in Likes																
%	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**Av**
0	-4	8	-1	-2	-6	-5	-1	-2	1	6	-2	7	0	-4	2	-3	-0.38
10	3	-8	10	1	-2	6	8	4	10	0	11	3	10	-1	-10	-1	2.75
20	8	-6	4	2	12	1	9	3	-7	3	4	2	15	1	4	-3	3.25
30	2	4	0	3	11	-4	-5	7	8	0	7	0	10	-5	0	13	3.19
40	4	2	10	3	7	16	1	3	4	7	5	11	4	3	4	0	5.25
50	4	5	2	6	8	11	6	11	2	-1	-1	13	1	8	11	10	6
60	6	1	6	4	-5	2	5	12	4	8	-2	11	4	15	6	11	5.5
70	14	8	4	10	11	6	5	-1	0	10	1	1	15	-2	3	1	5.38
80	4	2	3	2	9	2	14	6	-3	4	14	3	13	-5	-3	4	4.31
90	-3	6	16	11	9	7	3	9	8	1	7	8	12	-1	-5	-1	5.44
100	-1	6	10	13	2	15	10	6	18	1	11	-1	5	10	17	13	8.44



Variable Like Probability, Follow Probability 50%

	Trials. Difference in Likes																
%	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**Av**
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	2	1	0	3	3	2	1	-1	2	1	2	-1	2	-1	-2	1	0.94
20	-1	-1	-2	3	0	4	-3	1	-1	-1	1	4	-1	2	0	3	0.5
30	1	1	4	1	3	0	-2	0	-1	3	1	1	3	0	2	-1	1
40	12	7	-2	-4	5	-5	3	-6	-2	6	2	5	8	5	2	-3	2.06
50	2	8	4	9	12	5	1	3	11	-5	15	-1	5	7	-5	-5	4.13
60	3	11	2	4	12	11	0	3	4	8	-3	13	4	1	14	14	6.31
70	17	11	21	16	18	11	6	10	1	14	-4	9	7	12	4	4	9.81
80	11	12	8	11	5	20	12	5	8	9	12	6	4	2	6	5	8.5
90	9	8	7	12	3	3	2	4	10	0	5	13	6	0	2	0	5.25
100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



As it turns out, changing the like probability is more effective to a point, and then the difference in number of likes plummets. This is because, as we approach 100% liked, the difference in the first post and the last post is approaches 0, as it doesn't matter how many followers you have if every user will see the post due to liking anyway.

By looking at the trendline and the coefficient of determination for each graph R^2 , we can see that changing the follow probability is actually more effective than changing the like probability.
Like $R^2 = 0.291$
Follow $R^2 = 0.698$ (Stronger correlation)

Conclusion

While increasing like probability has a more significant effect to begin with, the plummet to zero after 70% makes it overall less significant according to the coefficient of determination.

This experiment was about finding the difference in subsequent posts from a user, if we were looking at the raw number of likes for any post as the probabilities changed, we would not have the 70% dropoff effect, and like probability would be a more significant metric than follow probability.

If I was to redo this experiment I would look into eliminating the non-variable probability while testing. This would eliminate a random variable and overall make the data more reliable. I would also look into further automating the data collection process to allow for more trials.