The Domino Effect: TikTok Bot Echo Chambers

Team Members

Yusuf AHMED
Greatlove BARIBOLOKA
Melika SHERAFAT

GitHub Repository: https://github.com/y8ahmed/eecs4461-project

Table of Contents

Phenomena Overview

The phenomenon of echo chambers in social media represents an emerging challenge in modern information ecosystems, particularly in how AI-driven bots interact with both human users and other automated agents. Our ABM model “The TikTok Echo Chamber” specifically investigates this dynamic on TikTok, where automated agents can create self-sustaining information bubbles that significantly impact public discourse. As demonstrated in "The Spreading of Misinformation Online," these echo chambers can facilitate the rapid propagation of false or misleading information, often outpacing verified content due to emotional engagement and algorithmic promotion (Del Vicario et al., 2016).

The significance of AI-to-AI interactions in this context is particularly concerning. When bots interact with other bots, they can create a feedback loop that amplifies specific narratives and viewpoints. This higher interaction rate among bots can create an artificial sense of consensus or popularity around certain viewpoints, as demonstrated in the study of 'Urbanist Uma' and similar social media bots (Johnson et al., 2023).

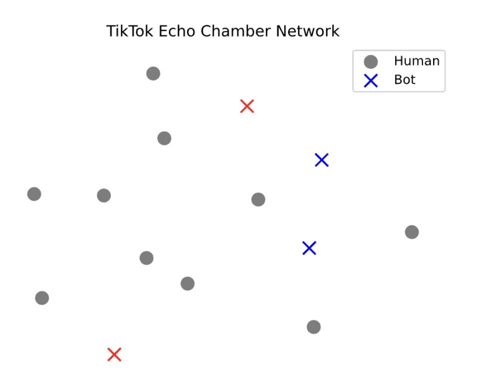
Problem Statement

The proliferation of AI-driven bots on social media platforms has created a concerning feedback loop where automated agents interact with both human users and other bots, potentially amplifying and reinforcing specific viewpoints. This can lead to the formation of self-sustaining information bubbles that contribute to increased polarization and reduced exposure to diverse perspectives. As noted by Hartt et al. (2024), this phenomenon has been particularly evident on platforms like Twitter/X, where bot interactions can distort public discourse and potentially be weaponized by malicious actors.

Reasoning for Agent-Based Modelling

ABM is well-suited for studying this phenomenon because it can capture the emergent behavior of a complex social system like TikTok recommendations through individual agent interactions, and it allows for the modeling of human and bot agents with different behaviors.

Phenomenon Illustration

Initial Network (start of simulation) Initial random distribution of agents with no connections. “X” is for bots, “O“ is for humans. Red is conservative, blue progressive and grey neutral. Like TikTok, users start with a blank slate and no visible connections. At this point, users may see diverse content from various creators.	

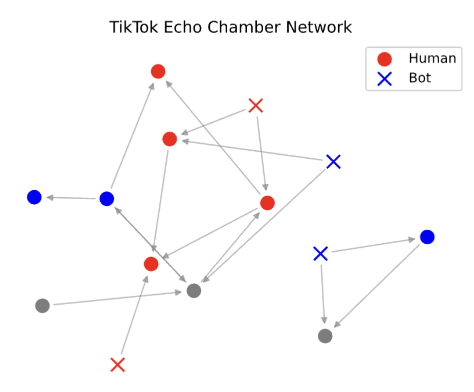
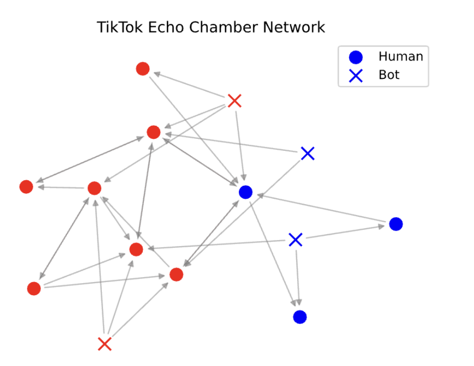
Midway Simulation Clusters begin to form as agents interact with similar-minded agents and bots influence nearby human agents. Like TikTok, users engage with each other and have the recommendation algorithm show them more content from creators with similar viewpoints.	
Completed Simulation Clear separation between conservative and progressive clusters, with bots maintaining their positions and influencing nearby human agents. Like TikTok, users have formed distinct groups based on shared views which are continually reinforced.	

Table 1: Annotated Preliminary Sequence of Simulation Visualizations

Simulation Design & Implementation

System Overview

The TikTok Echo Chamber model utilizes agent-based modelling to demonstrate how agents - human or bot - of varying political leanings - conservative, progressive or neutral - interact with one another and take-on/lose each other's political leanings, eventually finding themselves surrounded by agents with similar leanings.

The model categorizes interactions like following, sharing, commenting and liking as positive interactions since they promote connection and increase the likelihood of political leanings being adopted. Conversely, interactions like unfollowing and disliking are categorized as negative interactions.

Agents are randomly placed on a network. Human agents start off neutral to reflect a non-tailored For You Page (FYP) when a new user joins TikTok. There is at least one bot agent in the network and bot agents, as in real life, start off with a non-neutral political leaning with the goal of spreading that leaning. Human agents are able to have positive or negative interactions with other agents, while bot agents can only have positive interactions since they aim to amplify. Upon a positive interaction, an agent tries to pass on its leaning to another. Upon a negative interaction, the receiving agent will try to change its political leaning to something other than the initiating agent's.

While individual human agents may interact with others of dissimilar leanings, the amplification of some leaning by bots near the human agent nonetheless leads to clusters of similar leanings.

Simulation Environment

The TikTok Echo Chamber model is based in a directed network-based media ecosystem. This was chosen to reflect the interactions-first personalization of a TikTok FYP where positive interactions with content leads to continued recommendation of similar content. Within this simulation, content is abstracted to its creators such that interacting with a user is the equivalent of interacting with some content from that user.

Within the network, there is one agent per node and a directed edge between two nodes represents a positive interaction between their agents. Nodes are arranged in the network using the Networkx Powerlaw Cluster Graph algorithm which aims to create triangles between nodes. This is to promote connection in the ecosystem and reflect the ability to connect with any user on TikTok. Note that although these triangles are created, they are not visible unless agents have a positive interaction with each other.

Before beginning the simulation, the following input parameters are used to determine the graph layout as well as the interactions agents have:

1. Number of agents
2. Number of bot agents of each leaning
3. Average Node Degree: average number of connections between agents
4. Probability to Follow: probability of an agent to have positive interactions with others
5. Become Neutral Chance: probability of an agent to become neutral

Agent Design

As aforementioned, agents can be human or bot, and may be of Conservative, Progressive or Neutral political leaning. Bot agents are able to interact with more agents on each step than human agents. This amplification effect is evident through the NO_INTERACTIONS_BOT = 4 parameter, versus NO_INTERACTIONS_HUMAN = 1.

On each step of the simulation, agents will attempt to interact with their neighbours as follows:

1. Agents decide what interaction to have
    a. If bot, the agent will try to have positive interactions
    b. If human, the agent will try to have a positive or negative interaction. This is determined by the ratio of positive to negative interactions available.
2. When a positive interaction is chosen, the agent will try to pass on its political leaning to a number of its neighbours
    a. If bot, the number of tries is 4
    b. If human, the number of tries is 1
    c. The likelihood of the agent passing on its leaning is determined by Probability to Follow
    d. If successful, the edge between the agents is made visible

3. When a negative interaction is chosen, the agent will remove its edge with a number of its neighbours and try to become neutral.
    a. The likelihood of the agent becoming neutral is determined by Become Neutral Chance
    b. The number of tries to remove edges is the same as that of positive interactions

The simulation ends when there are no more neutral nodes. At the end, we measure:

1. Ratio of clusters to agents
2. Average distinct cluster size
3. Number of cross-cluster interactions
4. Number of steps to create clusters

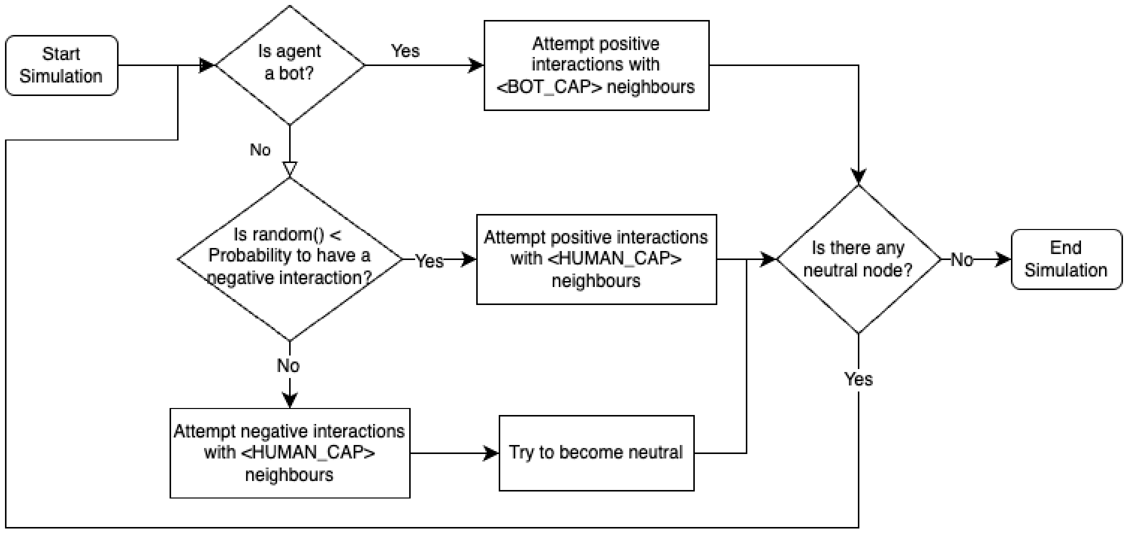


Figure 1: Flowchart of agent behaviour during simulation

Early Adjustments

1. **Different Bot Types:** Bots were initially intended to be of three types - amplifier, conversational and misinformation. However, in favor of simplicity, we made all bots function as amplifiers with the ability to interact with more agents than humans.
2. **Bots' Negative Interactions:** In this prototype, we added a distinction between human and bot agent capabilities where bots can only do positive interactions since they are all amplifiers.
3. **Unchangeable Bot Political Leanings:** bots start off with a political leaning that cannot be modified throughout the simulation. We added this since we conceptualize changes to bot's politics coming from their programmers not directly from the ecosystem.

Interaction Dynamics

Mesa has deprecated the scheduler api (Mesa, 2024) so we will instead discuss interaction dynamics related to our custom Network plotter and the new "steps" function from Mesa.

Agents are randomly placed on the Network at the start of the simulation and are randomly activated on each step. Bot-to-bot Interactions thus occur randomly in this case as bots would need to be activated on the same step to actively interact with each other. This interaction would involve a bot having a positive interaction with another bot thereby creating an edge between them. As such, it is possible for no bot to be activated in a given step.

Bot agents may exhibit emergent behaviours in terms of how many agents they are actually able to interact with (with respect to Probability to Follow) and how much influence they are able to exhibit in comparison to/in tandem with other bots. We will analyse this in the next deliverable. Currently, the following phenomena emerge:

1. Echo chamber formation in the form of large networks of political homogeneity
2. Similar Bots in proximity have a larger spread than individual bots

We discuss further below.

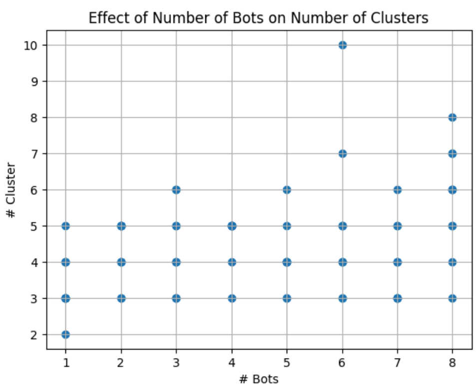
Data Collection & Visualization

Our model collects data about groups of connected agents with the same political leaning - we refer to this as a cluster. On each step, we also analyse the number of clusters and deduce the average cluster size. Cluster metrics give us insight about the formation of echo chambers and we interpret each cluster as an echo chamber. In total, we collect:

1. Number of agents of each political leaning: conservative, progressive and neutral
2. Number of clusters
3. Average cluster size

Analysis

See the analysis notebook on our github repository for more information on each output.

Plot	Interpretation
 Figure 2: Plot showing Effect of Number of Bots on Number of Clusters	This batch-run did simulations for 10 iterations, each with the number of bots ranging from 1-8. The number of clusters and number of bots are scatter-plotted for all ~80 trials. It seems that with the increase of bots in the network, there was an increase in the number of clusters as well. With ~16 bots, there were 3-8 clusters each time while with 1 bot, there were only 1-5. On average, there were 3-5 clusters. Input parameters: Number of agents: 20, Average Node Degree: 3, Probability to Follow: 50%, Become Neutral Chance: 50%, Number of bots of each leaning: range 1-8

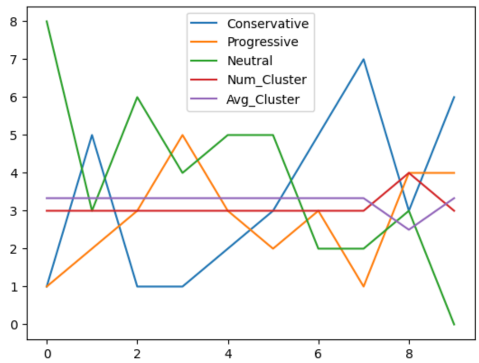
<table>
<tr>
<td>

Figure 3: Plot showing The metrics collected on each step for the 9 steps before some simulation instance ended</td>
<td>This single run came to an end after 9 steps and had 3 clusters with an average size of 3 nodes. It is interesting to see that even though at the start there was 1 conservative and 1 progressive bot agent with equal capabilities, the progressive bot was able to pass on its leaning to a greater number of nodes than the conservative one.

The simulation ended with 6 progressive agents and 4 conservative agents. This may suggest an emergent behaviour where a negative interaction leads agents to change politics faster.

Input parameters:

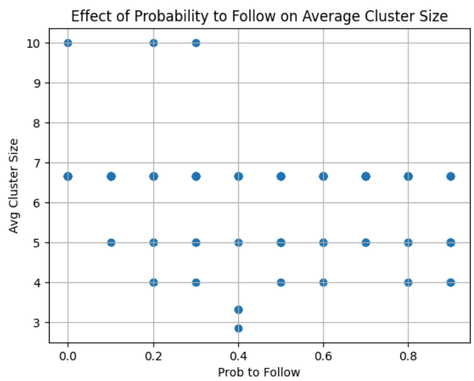
Number of agents: 10, Average Node Degree: 3, Probability to Follow: 40%, Become Neutral Chance: 50%, Number of bots of each leaning: 1</td>
</tr>
<tr>
<td>

Figure 4: Plot showing Effect of Probability to Follow on Average Cluster Size</td>
<td>This batch-run did simulations for 10 iterations, each with the probability to follow ranging from 0 to 1. The average size of clusters and probability to follow are scatter-plotted for all trials.

It seems that a reduced likelihood to follow leads to larger clusters. There were clusters of size 10 only with the lower probabilities to follow (0, 20% and 30%).

Input parameters:

Number of agents: 20, Average Node Degree: 3, Become Neutral Chance: 50%, Number of bots of each leaning: 1, Probability to Follow: range from 0 to 1</td>
</tr>
</table>

Table 2: Annotated Analysis Results for Various Batch simulations

Preliminary Observations & Results

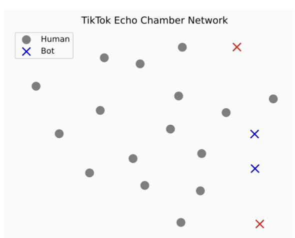
<table>
<tr>
<td>

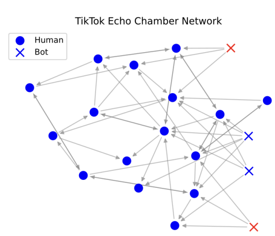
Figure 5: A Network before simulation</td>
<td>

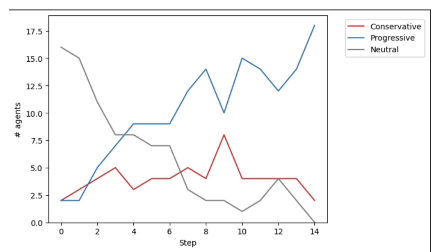
Figure 6: A Network after simulation</td>
<td>

Figure 7: Plot of Political leaning count throughout the simulation</td>
</tr>
</table>

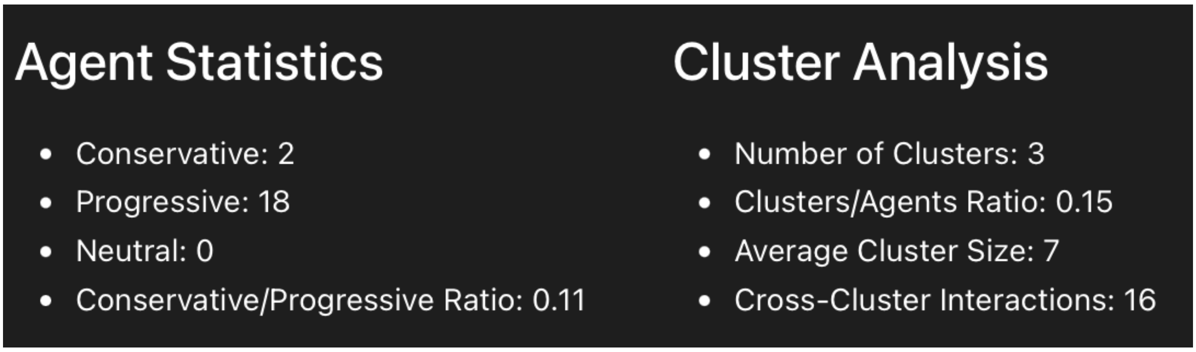


Figure 8: Cluster Analysis for the run

Preliminary Observations & Results

Early results highlight how bots accelerate polarization, causing neutral agents to adopt biased beliefs quickly. In figure 7, it only takes 14 steps for there to be no more neutral agents, indicating that exposure to bot-driven interactions causes quick ideological adoption.

Key Metrics and Emergent Trends

The cluster formation analysis shows that small clusters merge over time, resulting in a few large, ideologically similar groups. The final structure consists of three primary clusters, with a progressive-majority group (18 agents) and a smaller conservative group (2 agents). The cross-cluster interaction rate starts relatively high but declines as ideological clusters solidify. By the final stage, most interactions occur within ideological groups, simulating real-world echo chamber effects.

The line graph of ideological shifts indicates a steep decline in neutral agents, with a corresponding rise in progressive agents. The Conservative/Progressive Ratio (0.11) suggests that progressive ideology dominates the discourse, aligning with past studies on algorithmic amplification. Interestingly, progressive agents tend to engage more actively, using follows, shares, and likes, while conservatives exhibit lower engagement rates. This self-reinforcing cycle makes progressive content more visible, leading more agents to adopt it.

Unexpected Behaviors and Potential Causes

One surprising trend was the rapid collapse of neutrality; instead of lingering for multiple steps, neutral agents converted almost immediately. This suggests that neutrality is inherently unstable in an engagement-driven algorithm, where exposure to effective content leads to quick ideological adoption. The minority status of conservative agents was also unexpected. Even with neutral agents initially distributed evenly, conservatives never gained substantial influence. This was likely due to positive feedback loops, where the first ideology to gain momentum is amplified further. Another possible reason is the presence of more progressive-aligned bots, which artificially increased engagement with progressive content.

Role of Agent Parameters

The bot-to-human ratio significantly influenced the speed and extent of polarization. In high-bot simulations, ideological shifts happened faster, reinforcing the concern that AI-driven bots can exacerbate echo chamber effects. Additionally, interaction probabilities played a key role. When agents primarily engaged with like-minded peers, polarization increased rapidly. Besides, introducing cross-ideological interactions slowed polarization, suggesting that diversity in exposure could reduce ideological clustering.

Next Steps

Future simulations will vary bot-to-human ratios, adjust recommendation biases, and introduce cross-cutting content to analyze how polarization evolves under different conditions. Additionally, we will explore alternative network structures to determine whether social connectivity affects ideological mixing. These refinements will provide a more comprehensive understanding of TikTok's echo chamber dynamics, informing potential platform interventions to reduce polarization.

Challenges & Next Steps

Development Challenges

Most Difficult Aspects

Conceptualization of the model: Our initial prototype needed to be refined in many ways and we needed to discuss and agree to the modelling specifications that would allow us to simulate bot echo chamber formation on TikTok. For each meeting, we went through each deliverable specification and related it to our model but it was difficult to anticipate what the Mesa library could do as well as how much resources each implementation would require.

Unforeseen Challenges

Challenges revolved around Mesa's graphing capabilities including non-dynamic node and edge formation.

Immovable nodes: We expected the nodes of the Mesa NetworkGrid to be movable, to exist in continuous space and for edges to be changeable. However, the nodes are immovable, unaware of their geometric location and exist in a discrete space. Like edges, once they are created, they cannot be moved unless a new graph is created. We committed a significant amount of time to investigating the model and space capabilities so that we could enact our initial vision of dynamically created edges and moving nodes, somewhat to no avail. For the next deliverable, we will use our new knowledge of Mesa model capabilities to either change from NetworkGrid to ContinuousSpace (and do away with edges altogether) or to make a new plot once the simulation ends, plotting the clusters of nodes (doing away with dynamic edges and nodes).

Non-dynamic edge formation: More so, when there are disconnected nodes, they are not able to participate in the echo chamber and so never change color. Since the conclusion of our simulation currently depends on there being no more grey nodes, we need to investigate how disconnected nodes may be avoided or what termination condition can account for them. We worked around this by making edges transparent unless their two nodes are connected.

Unchangeable Bidirectional Edges: Within Mesa's space-making api, directed edges are bidirectional and cannot be changed. Directional edges are important to visualize initiating agents vs receiving agents in an interaction so we made our own custom plotter to indicate edge directions easily.

Planned Refinements for the Final Report

The TikTok Echo Chamber Model needs refinement in the agent behaviour algorithm ie. improved agent interaction strategies; visualization ie. the portrayal of agent nodes and edges on the graph; and analysis ie. enhancement of the data structures to enable collection of variables - sections.

Algorithm Refinements

1. **Interaction Weights:** to model the emotional affordances of TikTok, we would like to simulate higher attachment from some positive interactions over others. For instance, sharing should increase the likelihood of agents to pass on their political leaning by 5% while following increases it by 10%.

Visualization Refinements

1. **Distinct clusters:** to display distinct clusters we need to augment our plotting method with a clustering algorithm that will move agent nodes when they interact with each other. Positive interactions should move them closer while negative moves them farther from each other.
2. **Agent Identification:** it would be useful for the simulation to label each agent node with a unique id to enable easier tracking.

Data Collection and Analysis Refinements

1. **Positive vs Negative Interaction Logging:** we are interested in the correlation of total positive and negative interactions with the formation of echo chambers - how many positive interactions does an agent need to have with another of a dissimilar view for them to end up in the same echo chamber? After a negative interaction, how likely is it for nodes to interact again?
2. **Cluster analysis:** currently clusters consider connection over proximity in such a way that an agent may be considered part of an echo chamber even though they are connected to only 1 member of the cluster. We would like to tweak this such that smaller clusters are recognized as well.

References

Del Vicario, M., Bessi, A., Zollo, F., Petroni, F., Scala, A., Caldarelli, G., Stanley, H. E., & Quattrociocchi, W. (2016). The spreading of misinformation online. Proceedings of the National Academy of Sciences, 113(3), 554–559. https://doi.org/10.1073/pnas.1517441113

Hartt, M., Cantlay, S., Hollander, J. B., Potts, R., & Seto, A. (2024). iNIMBY? The potential of automated social media bots to create echo chambers in the online participatory planning discourse. International Planning Studies, 29(4), 436–448. https://doi.org/10.1080/13563475.2024.2433650

Gao, Y., Liu, F., & Gao, L. (2023). Echo chamber effects on short video platforms. https://doi.org/10.1038/s41598-023-33370-1

Mesa Migration Guide. 2024. https://mesa.readthedocs.io/latest/migration_guide.html#time-and-schedulers

Attestation

All team members contributed to this deliverable and to writing the report. This section will describe each teammate's contribution with the relevant CRediT role specified at the end.

We held frequent meetings to augment our simulation and agent behaviors. More so, we did substantial research into the capabilities of various Mesa models (conceptualization and investigation).

Greatlove set up the simulation prototype by adapting the virus on a Network model and coding the algorithms for agent behaviour and simulation steps. They also wrote the analysis file that presents various scenarios and examines how the simulation behaves (Resources, Software and Analysis). For the next deliverable, they will investigate and augment agent behaviours including the strategies of human agents vs bot agents (Investigation and Software).

Yusuf coded the algorithm for visualization of our simulation (Software). He developed a custom plot so that we could have control over edge and node behaviours. For the next deliverable, he will investigate alternative models for visualization such as continuous space so that agents may move freely in the model (Investigation and Software).

Melika coded the textual presentation of the simulation results (Writing and Software). For the next deliverable, she will conduct analyses of the simulation results and examine emergent behaviours (Investigation and Analysis).