LTM Enhancement for COVID-19 Modeling

Dylan Small
dylansmall@gatech.edu
Georgia Institute of Technology
United States

Tejas Pradeep
rptejas@gatech.edu
Georgia Institute of Technology
United States

Anjana Chamarthi
achamarthi6@gatech.edu
Georgia Institute of Technology
United States

1 ABSTRACT

The COVID-19 pandemic has shown the need for accurate disease spread models that account for individual-level variability in susceptibility. This project aims to develop a Linear Threshold Model (LTM) to simulate the dynamics of COVID-19 transmission within a population. The LTM represents individuals as dominoes with varying thresholds of influence needed to "fall" or, become infected. This approach tries to emulate the nuances of the human immune response, where some individuals succumb to infection after minimal exposure while others resist until more substantial contact occurs. The challenge lies in capturing the complexity of human mobility and interactions within the model. By integrating the LTM with other models such as the SIR model, we aspire to refine the predictive power of our model, offering a better understanding of disease spread systems.

2 RESPONSE TO MILESTONE COMMENTS

Our milestone feedback revealed some flaws in our project, and those flaws were paid special attention to in the final report. We have supported our work with added references to boost the foundational framework of the project. Further, we were able to combine the LTM model with an SIR model (and some variations of the two) to achieve significantly more accurate results closely following observed real-world data.

3 PROBLEM DEFINITION

Modeling the spread of disease has become increasingly difficult. As different people live different social lives, their susceptibility is variable. The objective of our project is to develop a Linear Threshold Model (LTM) that will accurately account for individual-level susceptibility to accurately model the spread of COVID-19 within a population. In the context of diseases like COVID-19, some people might get infected after minimal exposure, while others might require more significant contact with an infected person. The Linear Threshold Model (LTM) is the set of rules that tells us how much exposure each person requires to get infected. However, real life isn't as simple as a simulation. People move, interact in complex ways, and have different levels of immunity. So, the challenge is: Can we use the LTM to predict how a disease like COVID-19 spreads among people? And can we make this rulebook even better by combining it with other methods?

3.1 Formal Problem Definition

Objective: To enhance the Linear Threshold Model (LTM) for simulating the spread of the COVID-19 pandemic and to integrate it with more complex epidemiological models.

Problem Statement: Given a network of individuals where each node represents an individual and has an associated threshold value (indicating the level of influence required for the individual to adopt a behavior or become infected), the task is:

Simulation: Utilize the LTM to simulate the spread of the COVID-19 pandemic across the network. This involves determining the state of each node (infected or not infected) based on its threshold and the state of its neighbors.

Enhancement: Integrate the foundational principles of LTM with more intricate models like Susceptible-Infectious-Recovered (SIR) and Agent-Based Modeling (ABM). The goal is to develop a hybrid model that captures the strengths of LTM and the chosen complex model.

Validation: Compare the results of the LTM and the hybrid models with real-world data on the COVID-19 spread to assess the accuracy and reliability of the simulations.

Our group plans to research and incorporate the most helpful concepts from the Susceptible-Infectious-Recovered (SIR) model, Agent-Based Model (ABM), and Susceptible-Exposed-Infected-Recovered-Susceptible model (SEIRS) in order to accurately predict the dynamics of COVID-19 on a specific population.

4 PREVIOUS WORK

To properly understand the scope of our project and the models we will be using, we read some relevant literature on the subject of LTMs and other popular models out there. [8]

4.1 Inference of a Rumor's Source in the Independent Cascade Model

This paper discusses the Independent Cascade Model, a widely used framework for modeling rumor spreading and epidemic processes in networks [1]. In this model, a small set of nodes initially holds the rumor, and they have a probability of infecting their uninformed neighbors in discrete time steps. The primary focus of this work is on the inference problem, where the objective is to identify the source of the rumor or the "patient zero." This problem is crucial in various fields, including epidemiology and social network analysis. The authors present a maximum likelihood estimator for pinpointing the rumor's source based on a snapshot of active nodes after a certain number of time steps. They analyze this estimator's behavior in different types of networks, such as cycle-free graphs, d-regular trees, and Galton-Watson trees, showing that it undergoes a non-trivial phase transition as time progresses.

Furthermore, the paper explores the application of the Independent Cascade model in understanding information or influence propagation in networks, including its relevance in marketing, social media influence, and epidemiology. The authors emphasize the importance of inferring the source of information or disease outbreaks, highlighting its potential impact on preventing external influence on political elections and controlling the spread of diseases like COVID-19.

4.2 A Community Related Influence Evaluation Model for Social Networks with a Dynamic Perspective

This paper discusses the evolving dynamics of information diffusion and sharing within social networks (SNs), emphasizing the growing importance of community influence analysis [7]. While there has been substantial effort in evaluating the influence of individual users, this paper introduces a novel model called Dynamic Community Influence Evaluation Model (D-CIEM) to assess the influence of entire communities within SNs. D-CIEM distinguishes between two types of influence: entity influence and community-initiated influence, and it considers dynamic effects over time. The paper outlines the D-CIEM framework, and its formal definitions, and introduces dynamic entity influence evaluation methods, including user and community influence assessment with dynamic factors. Additionally, the paper presents methods for calculating community influence on internal and external users and other communities.

Social networks consist of individual users who form communities, and these communities play a substantial role in influencing users' interactions and information dissemination. However, analyzing community influence poses challenges due to dynamic member organizations and complex internal relationships. The paper argues that understanding community influence is vital, as it reflects the outcomes of user interactions within these communities. The research introduces a dynamic perspective to influence measurement, considering factors like time, membership changes, and community structural instability. It also categorizes communities into overlapping and non-overlapping structures for more comprehensive analysis.[15]

4.3 Linear Threshold Models to find influential people in a network

The paper titled "A fast algorithm for finding most influential people based on the linear threshold model" [11] by Khadje Rahimkhani aims to tackle the NP-Hard problem of finding the most influential nodes in a social network graph. The goal is to identify a subset of individuals (nodes) who can maximize the spread of influence through a target social network. The authors propose a new algorithm based on the linear threshold model (LTM) of influence maximization. The primary advantage of this algorithm is its ability to reduce the number of nodes investigated without compromising the quality, thereby decreasing its execution time.

The methodologies presented offer two notable benefits. First, the proposed algorithms are designed for enhanced speed and efficiency compared to current methods, making them invaluable for handling extensive datasets or networks, such as simulating the COVID-19 pandemic. Second, the paper explores the integration of LTMs with other models, providing insights on execution. This aligns directly with our project's primary objective, underscoring the paper's relevance and potential contributions to our work.

4.4 Solving Contamination Maximization on Networks for the Linear Threshold model

The paper addresses the challenge of minimizing the spread of undesirable entities, such as computer viruses and malicious rumors, by blocking a limited number of links in a network [4]. This optimization problem termed the contamination minimization problem, is a counterpart to the influence maximization problem, which seeks the most influential nodes in a social network for information diffusion. The authors of this paper adapted a method developed for the independent cascade model mentioned above to the linear threshold model.

The authors propose a method to find a good approximate solution to the contamination minimization problem under the linear threshold model. They adapted a greedy method developed for the independent cascade model to this problem. The method is based on a bond percolation process, which is a probabilistic model for the spread of contamination. The proposed approaches in this paper have a couple of significant advantages. Firstly, the proposed method is designed to be more efficient and run faster than previous traditional methods, making it very applicable to a large dataset such as the Covid-19 data set. Secondly, the paper provides a framework to analyze the LTM model against other approaches, which can be helpful when considering various approaches during this project.

4.5 Heterogeneous Peer Effects in the Linear Threshold Model

Typically, the Linear Threshold Model assumes that either individual thresholds are the same throughout the network, or they are randomly distributed. But this paper proposes causal inference methods that estimate individual thresholds [14]. This allows the model to more accurately predict if and when an individual will be affected by their peers. The paper introduces heterogeneous peer effects and thus reflects the idea that each peer asserts a different amount of pressure on their neighbors. In our case, this translates well to epidemiology. Each person has different immune systems and thus a different probability of contracting a disease. Each person also has a different social life and thus different probabilities of spreading disease. Using the heterogeneous peer effects this paper introduces, and the methods it uses to calculate these effects. We can take inspiration from this paper to incorporate diverse peer effects in our Linear Threshold Model.

The paper has impressive results with high threshold accuracy across multiple real-world based datasets. These datasets span from forest fires to hateful tweets, so we believe that since this paper draws impactful results from a variety of datasets, it will be applicable to disease spread as well. [12]

4.6 A Survey on Information Diffusion in Online Social Networks: Models and Methods

This paper serves as a reference point for our project, as it reviews important topics relating to online social networks [6]. The paper boasts the diverse set of problems that online social networks can be used to solve, and reviews the basic literature and future work that can be done within the intersection of explanatory and predictive models. The paper dives into explanatory models and their applications. The Susceptible-Infected model (SI), Susceptible-Infected-Susceptible model (SIS), SIR, and SIRS are explained with their strengths and weaknesses. Along that, the paper delves into individual and community influence and how they are ised in Online Influence Maximization (OIM) which is an expectation-maximization algorithm. This serves as a higher understanding of where the Linear Threshold Model came from, as it is a specific type of OIM. The paper also reviews different predictive models, such as the Independent Cascade Model (ICM), LTM, and Game Theory Model (GTM).

This paper provides us with a strong reference for a variety of current model literature in a basic way. If we are struggling to work with our current models, this paper serves as a good point to review options we have in our hand to better simulate the COVID-19 pandemic.

5 DATA COLLECTION

The dataset is structured to provide a weekly snapshot of the COVID-19 epidemiology figures across the United States. Each data point is identified by a unique key corresponding to an ISO 8601 date format (YYYY-MM-DD) signifying the start of the week for that datapoint. The collection process involves aggregating counts of new and cumulative confirmed cases, deaths, recoveries, and tests related to COVID-19. The total number of cases and deaths are monotonically increasing as time increases; we use the new cases and new deaths to create our infected and recovered population percentages for the various differential-equation based models.

It is noted that cumulative counts may not always equal the sum of daily counts. This discrepancy arises due to changes in counting criteria by health authorities and potential data omissions. Such factors cause the cumulative counts to diverge from the sum of daily counts over time. Therefore, cumulative values are reported in separate columns to maintain accuracy. This data was sourced from the CDC website found here.

6 MATH BACKGROUND

The mathematical background needed for LTM and SIR models largely revolves around a basic understanding of graph theory, optimization theory, and differential equations. The model definition outlines these mathematical concepts' applications to these models at a higher granularity. [5]

Monte Carlo Simulations

- **Purpose:** To handle stochastic processes in disease or information spread.
- **Python Libraries:** `numpy`, `matplotlib`, `seaborn`.

Dynamic Systems and Differential Equations

- **Purpose:** To model continuous dynamics in the SIR model.
- **Python Libraries:** `scipy.integrate`.

7 MODEL DEFINITION

The primary goal of this project is to create an LTM model to simulate and predict the outcomes of the Covid-19 pandemic. Implementing a successful LTM model is a multi-step process, with various design choices at each stage.The LTM model requires a systematic approach to captures the various dynamics of infection spread in the Covid-19 network. Below are some techniques we plan to use to accomplish this task.

- **Network Representation:**
  - *Graph Structure:* Represent the population or system as a graph $G(V, E)$ where V is the set of nodes (individuals) and E is the set of edges (relationships).
  - *Weighted Edges:* Assign weights to edges to represent the influence strength between nodes. These weights are derived from data representing the distance between the centers of each node, where each node is US state.
- **Influence Calculation:**
  - *Cumulative Influence:* For each node, calculate the cumulative influence from its neighbors.
  - *Activation Check:* If the cumulative influence on a node exceeds its threshold, it becomes infected in the next iteration.
- **Iterative Spread:**
  - *Initialization:* Start with an initial set of infected nodes. A set of active nodes is chosen to be the start point.
  - *Propagation:* In each iteration, check each inactive node. If its cumulative influence from infected neighbors exceeds its threshold, mark it for infection in the next cycle.
  - *Convergence:* Repeat the propagation step until no more nodes can be activated.

A secondary goal of this project is to combine an LTM model with established models like the SIR or ABM model to better understand and simulate disease spread. This task is more complex than the prior task and will require a more trial-and-error-based approach.

- **Model Layering:**
  - *Description:* Use each model to represent different layers or aspects of the problem. For instance, LTM is used to represent spread mechanics due to factors such as distance and migration, whereas SIR models infection spread.
  - *Technique:* Run each model separately but in a coordinated sequence, using the output of one model as input for another.
- **Hybrid Modeling:**
  - *Description:* Integrate the foundational principles of one model into another. For instance, incorporate the threshold dynamics of LTM into the state transitions of the SIR model.
  - *Technique:* Modify the differential equations of the SIR model to include LTM results, such as active nodes can spread more infection to their neighbors than inactive nodes.
- **Data-Driven Calibration:**
  - *Description:* Use real-world data to calibrate and validate the merged model, ensuring it accurately represents observed phenomena.
  - *Technique:* Adjust model parameters using optimization techniques or machine learning to fit the model to observed COVID-19 data.
- **SIR+ Models:**
  - *Description:* Experiment with various variations of the SIR model (SIR, SI, SIRS) to best model Covid-19

  - *Technique:* Implement the various models and compare their accuracies. We expect the SIRS model to be the most accurate due to the observed nature of Covid-19

8 ALGORITHMS AND TECHNIQUES USED

The key intuition behind this project is that the spread of a disease like Covid-19 cannot be accurately captured solely using an SIR model and infection statistics. A highly contagious disease like Covid-19 is influenced by many factors that cannot be captured in SIR statistics. We consider mobility and migration of people to be a key factor in the spread of Covid-19 and to back that claim we enhance SIR with a graph-based approach, using a graph of the United States (50 connected nodes) with distance as an edge weight. Further, this graph is enhanced using the LTM to better identify key nodes and help better model the propagation of the infection. While this assumption does not perfectly represent real-world scenarios, we believe it gives an approximation of real conditions and hence will help better capture infection spread dynamics. [3]

The below algorithms are used to implement the LTM model and to explore the combination of SIR with the LTM model. The algorithms are described on a high level without diving into implementation details.

Linear Threshold Model (LTM)

Graph Initialization.

- Define a directed graph $G = (V, E)$ with nodes V and edges E.
- Assign a weight w_{uv} to each edge $(u, v) \in E$, using the `networkx` library in Python.

Threshold Setting.

- For each node $v \in V$, assign a threshold θ_v randomly using the `numpy` library.

Activation Process.

- Initialize a set of active nodes A.
- For each inactive node v, sum weights from active neighbors divided by sum of weight from all neighbors. If the ratio exceeds θ_v, activate v.

Iteration.

- Repeat the activation process iteratively until no new activations occur.

Combining LTM with SIR

Model Integration.

- Integrate LTM thresholds and activations as a measure of infection rate. This is done as a multiplier on top of β

Network-Based SIR.

- Use the `networkx` library to represent the SIR model on a network, with LTM weights as transmission probabilities.

Behavioral Changes.

- Modify SIR rates based on LTM dynamics to reflect social influence on disease spread.

Hybrid Simulation.

- Simulate SIR dynamics and apply LTM activation checks using Python's `scipy.integrate` for differential equations.

Calibration.

- Use data to calibrate model parameters, employing machine learning techniques with libraries like `scikit-learn`.

Validation.

- Validate predictions with empirical data, adjusting the model with Python's data analysis tools like `pandas`.

9 MODEL EVALUATION

To ensure a comprehensive evaluation of our models, we will employ multiple metrics that capture different aspects of model performance. The primary metric will be a weighted mean-squared-error (MSE) loss, which will be complemented by normalized error metrics and correlation analysis.

9.1 Weighted Mean-Squared-Error Loss

The weighted MSE will serve as our primary loss function, giving us a measure of the average squared difference between the predicted and actual values, weighted by the importance of each prediction. The loss function is defined as:

$$L = \frac{1}{2N} \sum_{i=1}^{N} w_i \left[(y_i - \hat{y}_i)^2 + (z_i - \hat{z}_i)^2 \right] \tag{1}$$

Where y_i and z_i are the predicted population percentages of infections and deaths, respectively, $\hat{y}_i$ and $\hat{z}_i$ are the actual percentages, w_i are the weights assigned to each data point, and N is the number of data points.

9.2 Normalized Error Metrics

To account for the scale of the data and facilitate comparison across different models, we will calculate normalized errors such as the Root Mean Squared Error (RMSE)

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2} \tag{2}$$

These metrics will provide a normalized measure of the model's prediction accuracy, making it easier to compare performance across different datasets and models.

9.3 Visual Analysis

In addition to the error metrics discussed above, a visual test is critical to understand the models' accuracy. A good model shall resemble the real-world observed Covid-19 infection data throughout the observation period. Models that deviate significantly from the ground truth shall be discarded, even if they have good error scores.

9.4 Comparative Analysis

We will compare the loss, normalized errors, and correlation coefficients across multiple model experiments to determine which model combination simulates the COVID-19 data most accurately.

This multi-metric approach will allow us to capture a more nuanced picture of model performance and will allow us to tune parameters to best reflect observed data.

10 EXPERIMENTS & RESULTS

Our team chose to test various differential-equation based models with and without the conjunction of the Linear Threshold Model. We chose to test the performance of the SI, SIR, and SIRS models with and without the LTM graph as an underlying latent representation of the national COVID-19 data. The COVID-19 data was split into training and validation sets with a 80/20 split. We used grid search and `scipy.optimize` to find the optimal parameters for each model combination such that the training MSE loss is minimized. The performance is reported in terms of Mean-Squared Error (MSE) of the validation set's % of population infected ground truth.

10.1 LTM Graph Generation

We chose a 50-node, fully-connected graph to represent the United States. Each node represents a state, and the graph's edges between nodes represent the geographic distance between the two states. This way, the LTM prioritizes activating nodes with high citizen mobility. If a node is activated, we give it a higher chance to infect its neighboring nodes when running the differential-equation based models.

10.2 Results

Our team's goal is to find if the LTM provides enough latent information to the differential-equation based models for the validation MSE to decrease. The error can be thought of as the spatial squared distance between the ground truth and predicted infected values. Table 1 contains validation MSE scores for all model combinations.

	LTM	No LTM
SIR	0.0098	**0.0093**
SI	**0.00216**	0.0027
SIRS	**0.00138**	0.00757

Table 1: Validation MSE scores for various model combinations

From our experiments, we conclude that the latent representation of the United States via a Linear Threshold Model provides important information to properly calibrate and hyper-parameter tune the differential-equation based models that are augmented on top of it. Our team also wanted to investigate which simulation algorithm worked best with the LTM. From our results, the SIRS model in conjunction with the LTM allowed for a low MSE of 0.00138. Figure 1 contains the predicted and actual population percentages for this combination of models, and as shown the predicted curve closely models the ground truth. The SI model in conjunction with the LTM also provided a slight improvement. Overall, the factor that influenced our results the most was which differential-equation based model we chose. The SIR models don't pair well with the nature of COVID-19, and had high MSE scores when compared to the other two models. SI and SIRS were more realistic to modeling the behavior of COVID-19. In the SI model, infected patients have a chance to go back into the susceptible group, while in the SIRS model the infected patients either die and go into the "recovered" category, or transition into the susceptible group. The SIRS was most explanatory when the LTM was infused with it. We believe this to be because the latent representation of the country allowed for a slightly more complex model to be accurately paired with the national dataset. From our experiments, our team believes the LTM paired with the SIRS model most accurately simulates the COVID-19 pandemic within the United States. [2]

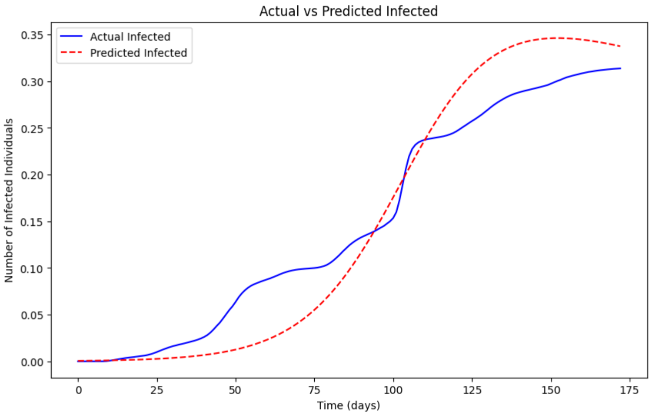

Figure 1: SIRS with LTM: actual infected population % v.s predicted infected population %. MSE = 0.00138

11 CHALLENGES AND NEXT STEPS

Data Accessibility and Usage

The project benefited significantly from the well-documented COVID-19 datasets provided by Google and the CDC. The comprehensive documentation and structured format of the data facilitated ease of use, allowing for a more streamlined data integration process within our simulation framework.

Challenges with LTM Implementation

Implementing the Linear Threshold Model presented several challenges:

- **Model Complexity:** The LTM's reliance on network structures introduces complexity, particularly in representing and managing the interactions between nodes within a vast network of individuals.
- **Threshold Dynamics:** Determining the appropriate threshold values for nodes and understanding how these thresholds influence the overall behavior of the system proved to be non-trivial, requiring experimentation and analysis using trial and error. [13]
- **Randomness:** Dealing with the inherent randomness in individual behavior and threshold activation necessitated the use of stochastic modeling techniques, which introduced additional layers of complexity in both the model development and outcome interpretation.

- **Simulation Validation:** Validating the simulation outcomes against real-world data was particularly challenging due to the stochastic nature of the model. Ensuring that the model reliably replicates the dynamics of COVID-19 spread required meticulous calibration and verification processes.

11.1 Conclusions and Future Work

Through this project, we aim to test the accuracy of standard models such as a SIR and the effect augmenting the model with real-world data shall have on accuracy. Through our experiment we have shown augmenting the SIR model with real-world data can significantly increase its accuracy and help it better resemble the real world. When modeling a complex disease such as Covid-19, statistics on infections and recoveries are not enough to capture the disease's dynamics. Augmenting such statistics with data regarding indirect factors that can influence spread and help a model paint a more realistic picture. Our model combined the SIRS model with an LTM model run on a graph representing the distance between US states, and while that is a simple factor that can influence disease spread, that simple addition helped the model be significantly more realistic. [10]

This project was a great learning experience in the dynamics of epidemiology and disease spread prediction. Further, this project helped highlight some of the difficulties in the field, regarding working with real-world data and handling the un-predictable behavior of a disease. Future work on this project can take multiple routes to enhance the model. One approach can be to use an enhanced dataset to better capture mobility between states. A dataset detailing the likelihood of travel between states or the number of flights between states may better capture mobility. Further, improvements can be made to the SIRS model to add in a factor to delay re-infections due to immunity provided by a prior infection. This may help better model the observed nature of Covid-19. Lastly, incorporating data such as vaccination data can help model yet another element of the pandemic and can help gain insights on how and where vaccinations can be best deployed. [9]

REFERENCES

[1] Petra Berenbrink, Max Hahn-Klimroth, Dominik Kaaser, Lena Krieg, and Malin Rau. 2022. Inference of a Rumor's Source in the Independent Cascade Model. (2022). arXiv:2205.12125 [cs.SI]

[2] Wei Chen, Yifei Yuan, and Li Zhang. 2010. Scalable Influence Maximization in Social Networks under the Linear Threshold Model. (2010), 88–97. https://doi.org/10.1109/ICDM.2010.118

[3] Charanpal Dhanjal and Stéphan Clémençon. 2014. An SIR Graph Growth Model for the Epidemics of Communicable Diseases. (2014). arXiv:1312.2565 [stat.AP]

[4] Masahiro Kimura, Kazumi Saito, and Hiroshi Motoda. 2008. Solving the contamination minimization problem on networks for the linear threshold model. (2008), 977–984.

[5] J.C. Lang and H. De Sterck. 2016. A hierarchy of linear threshold models for the spread of political revolutions on social networks. *Journal of Complex Networks* 4, 3 (2016), 426–456. https://doi.org/10.1093/comnet/cnv030

[6] Mei Li, Xiang Wang, Kai Gao, and Shanshan Zhang. 2017. A Survey on Information Diffusion in Online Social Networks: Models and Methods. *Information* 8, 4 (2017). https://doi.org/10.3390/info8040118

[7] Meizi Li, Yang Xiang, Qian Zhang, and Bo Zhang. 2018. A community related influence evaluation model for social networks with a dynamic perspective. *Mathematical Problems in Engineering* (Jul 2018). https://doi.org/10.1155/2018/8697275

[8] Yoshihiro Maki and Hideo Hirose. 2013. Infectious Disease Spread Analysis Using Stochastic Differential Equations for SIR Model. (2013), 152–156. https://doi.org/10.1109/ISMS.2013.13

[9] Nishith Pathak, Arindam Banerjee, and Jaideep Srivastava. 2010. A Generalized Linear Threshold Model for Multiple Cascades. (12 2010), 965–970. https://doi.org/10.1109/ICDM.2010.153

[10] Patryk Pazura, Jarosław Jankowski, and Kamil Bortko. 2020. Modeling the Performance of Limiting Information Spreading Processes under Competing Linear Threshold Model. *Procedia Computer Science* 176 (2020), 3751–3760. https://doi.org/10.1016/j.procs.2020.09.012 Knowledge-Based and Intelligent Information Engineering Systems: Proceedings of the 24th International Conference KES2020.

[11] Khadije Rahimkhani, Abolfazl Aleahmad, Maseud Rahgozar, and Ali Moeini. 2015. A fast algorithm for finding most influential people based on the linear threshold model. *Expert Systems with Applications* 42, 3 (2015), 1353–1361.

[12] Ahmad Sedaghat, Shahab Band, Amir Mosavi, and Laszlo Nadai. 2020. Predicting COVID-19 (Coronavirus Disease) Outbreak Dynamics Using SIR-based Models: Comparative Analysis of SIRD and Weibull-SIRD. (2020), 000283–000288. https://doi.org/10.1109/CANDO-EPE51100.2020.9337791

[13] Ashis Talukder, Md. Golam Rabiul Alam, Nguyen H. Tran, Dusit Niyato, Gwan Hoon Park, and Choong Seon Hong. 2019. Threshold Estimation Models for Linear Threshold-Based Influential User Mining in Social Networks. *IEEE Access* 7 (2019), 105441–105461. https://doi.org/10.1109/ACCESS.2019.2931925

[14] Christopher Tran and Elena Zheleva. 2022. Heterogeneous Peer Effects in the Linear Threshold Model. *CoRR* abs/2201.11242 (2022). arXiv:2201.11242 https://arxiv.org/abs/2201.11242

[15] Degang Xu and Xiyang Xu. 2014. Modeling and control of dynamic network SIR based on community structure. (2014), 4648–4653. https://doi.org/10.1109/CCDC.2014.6853003