An LLM-enhanced Agent-based Simulation Tool for Information Propagation

Yuxuan Hu\textsuperscript{1}, Gemju Sherpa\textsuperscript{1}, Lan Zhang\textsuperscript{2}, Weihua Li\textsuperscript{2}, Quan Bai\textsuperscript{1}, Yijun Wang\textsuperscript{1} and Xiaodan Wang\textsuperscript{3}

\textsuperscript{1}University of Tasmania, Hobart, Australia
\textsuperscript{2}Auckland University of Technology, Auckland, New Zealand
\textsuperscript{3}Yanbian University, Jilin, China
\{yuxuan.hu, sherpag, quan.bai, ywang134\}@utas.edu.au, \{lan.zhang, weihua.li\}@autuni.ac.nz, 0000007458@ybu.edu.cn

Abstract

Influence diffusion models are used for simulating information propagation in social networks. While most existing influence diffusion models are probabilistic, the emergence of Large Language Model (LLM) sheds light on the language-level inferences and interactions of user agents. This paper presents an LLM-enhanced Agent-based Influence Diffusion model (LAID), and a web-based visualization tool, LAIDSim, for simulating the information propagation in social networks.

1 Introduction

Information propagation describes how pieces of information, such as news, ideas, opinions, and advertisements, spread through social networks [Gomez-Rodriguez et al., 2013; Wu et al., 2019]. Simulating this phenomenon aids in understanding how information diffuses through these networks. Influence diffusion models [Kempe et al., 2003], extensively used to simulate information propagation, follow a ‘contagion’ pattern based on the network structure [Gomez-Rodriguez et al., 2013], where influence spreads from one node to another through connecting edges. Agent-based modeling (ABM) [Macal, 2016] has been applied to simulate influence diffusion. By leveraging the advantages of ABM [Li et al., 2023; Li et al., 2016; Hu et al., 2019], the diffusion process is driven and captured by the behaviors of individual agents, allowing for a more accurate representation of the complex and dynamic nature of information diffusion.

Most existing influence diffusion models are probability-based, which presents several challenges. Firstly, the content of information plays a critical role in individuals’ decision-making behaviors, but is often ignored by most probability-based models. Although some topic-aware influence diffusion models [Chen et al., 2015] consider how tagged content impacts these probabilistic models, they are typically constrained to predefined topics [Chen et al., 2023]. Secondly, probability-based models, including agent-based models, neglect individuals’ ability to alter information [Wang et al., 2023]. Specifically, the content of a piece of received information can be altered according to an individual’s prior knowledge or interests.

In recent years, the emergence of Large Language Models (LLMs) has shed light on filling some of the above gaps [Gao et al., 2023]. In this paper, we present an LLM-enhanced agent-based influence diffusion model, named LAID, along with a software simulation tool, named LAIDSim, for simulating influence diffusion. LAID is developed based on the conventional Independent Cascade model [Kempe et al., 2003], while leveraging ABM [Li et al., 2023] to capture the personal traits and individual behaviors of social network users. In LAID, users are represented as autonomous agents with distinctive profiles and complex social behaviors. Information content is considered a crucial factor influencing user agents’ decision-making behaviors, specifically whether to be influenced by received messages. We particularly emphasize information alteration [Wang et al., 2023] as a key feature in the propagation process, referring to the ability of user agents to modify the content of initially spread information during peer-to-peer propagation. LLMs are utilized to realize this feature, serving as the ‘brain’ of user agents. The LLM generates the content of an influence message based on the user agent’s profile and the content of its received message, facilitating further propagation to its neighbors.

LAIDSim is implemented based on the ABM framework Melodie [Yu and Hou, 2023]. This tool enables end-users to simulate information propagation with self-defined scenarios, visualize the propagation process over time, and generate analytical results at a global level. To demonstrate the capabilities of LAIDSim, we designed and conducted four scenarios within two experiments. These experiments showcase the advantages of LAID in representing language-level behaviors of social network users and highlight the ability of LAIDSim to simulate information propagation across various application settings.

2 The LLM-enhanced Agent-based Influence Diffusion Model

2.1 Preliminaries

In LAID, a social network is modelled as a directed graph $G = (V, E, M)$ with a set of user agents $V = \ldots$

1https://youtu.be/AaxDqK96RZE
2https://github.com/shaunahu/LAIDSim
At the beginning of the influence diffusion, we randomly selected to spread an influence message $m_s$ to their neighbors.

### 2.4 Influence Diffusion

Influence diffusion is developed based on the conventional IC model. In the IC model, each user agent has two possible states: active and inactive. An inactive user agent has a single chance to be activated by its active neighbors with an influence probability. The diffusion process stops if there were no more user agents could be influenced.

The influence probability $p_{ij}^k$ is the probability for a user agent $v_i$ to accept an influence message $m_k$ from its neighbor $v_j$. This is calculated by the consideration of the content of the message $m_k$ and the relationship between two user agents:

$$
p_{ij}^k = \alpha \cdot \text{sim}(m_k, v_i) + (1 - \alpha) \cdot \text{sim}(v_i, v_j).
$$

In Equation 1, $\text{sim}(m_k, v_i)$ is the cosine similarity between the content embedding $c_k$ of $m_k$ and the embedding $r_i$ of $v_i$. It suggests how the content $c_k$ of an influence message $m_k$ matches a user agent $v_i$:

$$
\text{sim}(m_k, v_i) = \frac{c_k \cdot r_i}{\|c_k\| \cdot \|r_i\|}.
$$

Similarly, the similarity between two user agents $v_i$ and $v_j$ is measured by the cosine similarity between the user profile embeddings $r_i$ and $r_j$, where $r_i$ and $r_j$ are the profiles of $v_i$ and $v_j$:

$$
\text{sim}(v_i, v_j) = \frac{r_i \cdot r_j}{\|r_i\| \cdot \|r_j\|}.
$$

In the influence diffusion process, we identify content influence as a critical factor in individuals accepting an influential message, denoted as $\text{sim}(m_k, v_i)$. The influence parameter $\alpha$ in the proposed diffusion model is a trade-off between these two factors. A higher $\alpha$ indicates that the user agent is more likely to be influenced by social influence rather than content, while a lower $\alpha$ suggests that the user agent is more sensitive to the content of the information but pays little attention to its neighbors’ decisions.

Once a user agent is influenced, it accepts the spreading influence message $m_k$ and further propagates to its neighbors. Specifically, LLM model is adopted for information alteration. That is, a user agent can alter the content of an existing influence message $m_k$ that it received from any of its in-neighbors $v_j \in \Gamma_{in}^i$, adding or changing words based on its profiles, and further diffuses the altered influence message $m_k'$ to its out-neighbors $\Gamma_{out}^i$.

### 2.5 Information Alteration

GPT-3.5 Turbo is used in this stage for user agents to generate altered influence messages based on their profiles and the influence messages they have accepted. To evaluate the changes between two influence messages, we adopted alteration degree [Wang et al., 2023] to measure the extent of information alteration by a user agent.
### Table 1: Scenarios with different message contents and application settings

<table>
<thead>
<tr>
<th>ID</th>
<th>Content</th>
<th>Application Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Meet your local election candidates: [Candidate 1] focuses on education reform and community safety. [Candidate 2] advocates for sustainable development and environmental initiatives. [Candidate 3] prioritizes small business support and infrastructure improvements. Get informed, attend candidate forums, and choose who aligns with your vision for our community’s future. #LocalElections</td>
<td>Political election</td>
</tr>
<tr>
<td>2</td>
<td>Introducing our latest product: The SmartHome Hub! Seamlessly control all your devices, from lights to thermostats, in one central hub. Enjoy energy savings, convenience, and security. Compatible with popular smart home platforms. Upgrade your home today and experience the future of intelligent living. Available now at authorized retailers.</td>
<td>Viral marketing</td>
</tr>
<tr>
<td>3</td>
<td>Introducing a novel concept: Shared Skill Spaces. A communal area where diverse talents intersect, fostering collaboration and innovation. Unleash creativity, maximize collective expertise, and elevate the potential of our community. Explore Shared Skill Spaces today!</td>
<td>Spread new idea</td>
</tr>
<tr>
<td>4</td>
<td>How are you today?</td>
<td>Daily conversation</td>
</tr>
</tbody>
</table>

### 3 Experiments

#### 3.1 Experimental Settings

**Parameters:** We consider seed size \( k \) and influence parameter \( \alpha \) as two parameters of our proposed model that can be adjusted from the front-end. The seed size \( k \) represents the number of initially active user agents. A higher \( k \) suggests a quicker diffusion but also entails higher costs. As explained in Subsection 2.4, the influence parameter \( \alpha \) potentially impacts the influence probability by controlling the trade-off between social influence and content influence.

**Scenarios:** We designed four scenarios with varying initial influence message content and application settings for the proposed model to simulate. The initial influence message is also generated by the GPT-3.5-Turbo model. Table 1 lists these scenarios.

**Initialization Setup:** At the beginning of each scenario simulation, a synthetic network based on Erdos-Renyi model [Erdős et al., 2012] is randomly constructed with a size of 20 nodes, and a probability of 0.1 for these nodes to connect to each other.

#### 3.2 Experiment 1: Influence Coverage Analysis

This experiment analyses the cumulative number of user agents in two states: active and inactive over a particular time period, i.e., 10 time steps. Figure 2 shows the simulation results. The slight differences of the numbers of active and inactive agents across the four scenarios suggest that user agents possess the capability to respond to information with different contents. In all four scenarios, the diffusion process adheres to a contagion pattern, indicating LAID’s capacity to effectively simulate information propagation.

#### 3.3 Experiment 2: Information Alteration Analysis

Figure 3 shows the simulation results of four scenarios monitored from time step 1. It can be seen that all four scenarios have been slightly altered at the first time step compared to the initial influence message. \( S_2 \) shows the most significant alteration. This is because the initial influence message in \( S_1 \) is an ‘open’ question. The alteration degree scenarios \( s_2 \) to \( s_4 \) decreases, implying that user agents generate information with a similar pattern to their received influence messages. The degree of alteration converges after a few time steps, indicating that the diffusion process stops, and no more user agents can be influenced to generate new influence messages.

### 4 Conclusion

In this paper, we presented an LLM-enhanced agent-based simulation tool, LAIDSim, for simulating information propagation in social networks, based on our proposed diffusion model LAID. We showcased the interpretability of the proposed model across four scenarios by demonstrating language-level interaction and inference of user agents. In the future, we plan to extend our work by improving the user agent model with more comprehensive user behaviors.

**Contribution Statement**

The first two authors contributed equally to this work and should be considered as joint first authors.

**References**


