

**Universidad de La Habana**

FACULTAD DE MATEMÁTICAS Y CIENCIAS DE LA COMPUTACIÓN

# SIMULACIÓN DE PROPAGACIÓN EN REDES SOCIALES

Integrantes:

Alex Sánchez Saez C412

Carlos Manuel González C411

Jorge Alberto Aspiolea González C412

Abril 2024

# 1. El problema

La presente investigación consiste en extraer de una red social, simulando el comportamiento básico de las personas en la misma, darle una respuesta a la pregunta que nos planteamos que es: *¿Dado una red de seguidores en una red social, cual es el tipo de post que más alcance tiene en dicha red?* Para resolver esta pregunta vamos a realizar una simulación de dicha red social.

## 2. Simulación

La simulación consiste en un *medio ambiente*, el cual sería nuestra red social. Este estará representado por un conjunto de posts que abarca diversos temas, formando un vector de características, en el cual, que tan relevante es cada tema en ese post. También los posts serán evaluados por tres estadísticas importantes, *la cantidad de likes, la cantidad de dislikes y la cantidad de veces que se compartió ese post*.

Por otra parte, las personas serán representadas agentes los cuales podrán tomar decisiones y decidir cómo reaccionar a los post, pudiendo decir entre dar like, dislike o compartir la publicación con otros. Para modelar este sistema de agentes usaremos una **arquitectura multiagentes** en donde los agentes podrán interactuar entre ellos y con el medio ambiente. Estos agentes actuarán basados en la arquitectura **Believes-Desires-Intentions(BDI)**

## 3. Optimización

Una vez la simulación arroge las estadísticas antes mencionadas, ahora tenemos la interrogante de encontrar *¿qué combinación de temas y en qué cantidad proporciona un mejor alcance en la red?*. Esta interrogante tiene a dos pasos importantes; primero es ¿cómo calificas que una publicación proporciona mayor alcance? ¿bajo que criterio es interesante realizar una búsqueda de la mejor solución? y en segundo lugar tenemos el problema de *¿Cómo vamos a encontrar dicha solución óptima?*

### 3.1. Criterios de optimalidad

Para responder la primera pregunta, podemos ver que cada post en la red social se mide por tres estadísticas principales: *cantidad de likes, cantidad de dislikes y veces compartida*, además que la simulación nos proporciona esas mismas estadísticas enfocándonos esta vez en el tipo de tema, por tanto tendríamos un índice de que tanto gusta cada tema entre los usuarios (cantidad de likes promedio), o cual es el tema que menos gustó (cantidad de dislikes promedio), o incluso cuáles son los temas que más se difundieron, ya sea por gustos o disgustos, en la red (cantidad de veces compartida). Entonces depende de los intereses del tengamos podemos decidir que importancia darle a cada una de esas tres estadísticas.

Para esto lo más interesantes es que, en vez de tener que definir numéricamente que tanto nos interesa una estadística más que otra, definamos en lenguaje natural cuales son nuestras intenciones respecto al tipo de alcance que queremos obtener y de ahí, usando un **LLM** extraer los índices numéricos que más relación tenga con nuestra descripción.

### 3.2. Encontrar la mejor solución

Para la segunda pregunta de *¿cómo encontrar la mejor solución?*, debemos fijarnos que la solución consiste en un vector  $x$  de tamaño  $n$ , donde cada componente del vector es un número  $x_i \in [0, 1]$  que indica que el tema  $i$  debe tener una relevancia  $x_i$  en el post para que tenga el mayor alcance.

Para saber si una solución es mejor que otra debemos tener una forma de cuantificar dicha mejora y así poder hacer la comparación. Para esto definimos una función  $f : D \rightarrow \mathbb{R}$ , donde  $D$  es el dominio de todas las soluciones posibles del problema, o sea, todas las formas posibles en las que se puede construir un post, y  $f(x) \in \mathbb{R}$  un número que indica que tan buena es la solución. Podemos ver que  $D$  es de tamaño infinito, ya que una solución consiste en una distribución de  $n$  números entre 0 y 1, y como ese conjunto es no enumerable entonces hay infinitas combinaciones de  $n$  números que pertenezcan al rango  $[0, 1]$ . PO

Por ende, una buena forma de enfrentar el problema es usando metaheurísticas para encontrar a la función objetivo  $f$  un óptimo que se aproxime a la solución real. Como metaheurística usamos **Particles Swarm Optimization (PSO)**, la que es buena para optimizar funciones.

### 3.3. Función de optimización

En la simulación existen  $n$  características que describen los diferentes temas de los que puede tratar un post. Cada post en la red social tiene un vector  $v$  de  $n$  dimensiones con valores  $v_i \in [0, 1]$  que indica que tan relevante es el tema  $i$  en el post. También, de la simulación se extrae para cada característica, tres indicadores importantes (Likes, dislikes y veces compartida). Por tanto, podemos crear una matrix  $C_{n,3}$  donde  $C_{i,j}$  indica que porcentaje de la red tuvo la reacción  $j$  en posts donde el tema  $i$  es relevante.

El objetivo es entonces encontrar el vector  $v$  tal que tenga la mejor combinación de relevancias por cada característica y nos dé el mayor crecimiento en la red. Como el impacto de una post de un tema  $i$  se mide por los 3 índices expuestos anteriormente ('likes', 'dislikes', 'shared'), entonces necesitamos un valor que indique cuan relevante es este índice para un post. Sea  $\alpha, \beta, \lambda \in [-1, 1]$  los respectivos índices de relevancia, en donde  $-1$  afecta muy negativamente al post y  $1$  afecta muy positivamente al post. Por otro lado para medir el impacto de una publicación en la red esta se podría calcular como:

$$I(v) = \alpha g_1(v) + \beta g_2(v) + \lambda g_3(v)$$

donde  $g_1(v)$ ,  $g_2(v)$  y  $g_3(v)$  indican cuanto afectan los índices de crecimiento respectivamente entre todos los temas.

La forma de calcular las 3 funciones son la misma, solo que se separan en funciones diferentes para mejor comprensión. Para esto hacemos uso de la función exponencial para recompensar las a los valores mas grandes de  $x$ . Entonces podemos definir la función  $g_1(v)$  que indica cuanto ... como:

$$g_1(v) = v_1 \alpha e^{\sum_{i=1}^n C_{i,1}}$$

Entonces la función en su totalidad la podemos expresar como:

$$I(v) = \alpha g_1(v) + \beta g_2(v) + \lambda g_3(v) = v_1 \alpha e^{\sum_{i=1}^n C_{i,1}} + v_2 \beta e^{\sum_{i=1}^n C_{i,2}} + v_3 \lambda e^{\sum_{i=1}^n C_{i,3}}$$

Si definimos el vector  $z = [\alpha, \beta, \lambda]$  entonces la función quedaría:

$$I(v) = \sum_{j=1}^3 z_j v_i e^{\sum_{i=1}^n C_{i,j}}$$

Ahora bien, como queremos hallar el vector  $x$  tal que maximize el impacto en la red entonces tenemos que optimizar la funcion  $I$  para calcular el máximo:

$$\max I(v) = \max \sum_{j=1}^3 z_j v_i e^{\sum_{i=1}^n C_{i,j}}$$