Aproximación computacional a dinámicas de opinión en redes sociales

Carlos Ramírez Camilo Rocha

Árboles y Grafos 2023-2: Proyecto de curso

1. Introducción

Las redes sociales se han convertido en una parte imprescindible de nuestras vidas diarias. Plataformas como Twitter (ahora X), Facebook, Instagram y TikTok, entre otras, permiten la comunicación entre usuarios de diferentes partes del mundo. También permiten publicar aspectos de la vida diaria de cada quien, incluyendo ideas, fotos y videos.

Un beneficio de las redes sociales es su capacidad para conectar comunidades de usuarios, i.e., personas con intereses comunes. Las redes sociales pueden ayudar a cambiar la opinión de sus usuarios. Por ejemplo, cuando buscan nueva información sobre un tema en particular. Además, cuando un usuario tiene una opinión clara acerca de un tema, este usuario prefiere la información que proviene de grupos que confirman su creencia inicial. También se ha encontrado que quienes prestan mayor atención a agentes externos (e.g., compañeros de estudio, familiares, figuras públicas, artistas, líderes religiosos o culturales) tienden a cambiar su opinión con mayor probabilidad o frecuencia. Es decir, dependiendo de la influencia de agentes externos, algunos usuarios pueden ser más propensos a cambiar de opinión acerca de temas. Este fenómeno ha sido particularmente nocivo cuando se han usado redes sociales para polarizar a la sociedad en el marco de elecciones democráticas.

La intención de este proyecto es utilizar el marco de los árboles y grafos, junto con algoritmos sobre ellos, para modelar, simular y analizar las dinámicas de opinión de usuarios en redes sociales.

2. Planteamiento del problema

Una red social se modela como un grafo dirigido G con vértices V y arcos E. Los vértices corresponden a los usuarios de la red y los arcos a las relaciones binarias y dirigidas entre dos usuarios: un arco (u, v) indica que el usuario u tiene influencia sobre la opinión de v. Las opiniones están asociadas a los usuarios de una red social. Para el propósito de este proyecto se supondrá que las opiniones de los usuarios giran en torno a un tema dado y que estas se miden con un número real en el rango $[0..1]$. Este valor representa el nivel de asentimiento de un usuario con respecto al tema. Así, el valor 1 representa que el usuario está completamente de acuerdo y el valor 0 representa que el usuario está completamente en desacuerdo. En el marco del modelo de grafos, la opinión de los usuarios en un momento dado corresponde a una función $\alpha : V \to [0..1]$ de tal manera que para cualquier usuario u, la expresión $\alpha(u)$ denota la opinión de u. De manera similar, el peso de la influencia de un usuario u sobre un usuario v se modela con una función $\beta : E \to [0..1]$. De esta manera, dado un arco (u, v), $\beta(u, v) = 1$ representa que el usuario v es completamente influenciable por el usuario u y $\beta(u, v) = 0$ representa que la influencia de u sobre v es nula. Puede suponer que no hay arcos de un vértice a si mismo.

El problema general que trata este proyecto se especifica de la siguiente manera:

Entrada: $G = (V, E)$, $\alpha : V \to [0..1]$, $\beta : E \to [0..1]$ y $N \geq 0$.

Salida: La función $\alpha_N : V \to [0..1]$ indicando la opinión de los usuarios V después de interactuar N veces.

Para que el problema general pueda ser entendido es necesario explicar varias cosas. Primero, qué se entiende por una interacción. En este caso, una *interacción* es la actualización simultánea de la opinión de un subconjunto de usuarios. Por ejemplo, si los usuarios son $0, 1, 2, 3, 4$, en una interacción se pueden actualizar las opiniones de 1 y 2 como consecuencia de la influencia de algunos de los usuarios con los que se relacionan, y en otra las opiniones de 2, 3 y 4 también con respecto a la influencia de algunos de sus conocidos o amigos. Segundo, la forma en que las opiniones se actualizan. Para ello, es necesario definir una fórmula que indique cómo la opinión de un usuario cambia con base en las opiniones ponderadas de algunos de los usuarios que lo influencian. Sea $u \in V$, $E_u \subseteq E$ el conjunto de arcos incidentes a u y $A \subseteq E_u$. Además, suponga que $V_A = \{v \mid (v, u) \in A\}$ el conjunto de vértices incidentes a u con arcos en A. La opinión actualizada de u con respecto a A, denotada $\alpha^A(u)$ se define como:

$$\alpha^A(u) = \alpha(u) + \begin{cases} 0 & , |V_A| = 0, \\ \frac{(+v|v\in V_A:\beta(v,u)(\alpha(v)-\alpha(u)))}{|V_A|} & , |V_A| \neq 0. \end{cases}$$

3. Primera parte

En esta parte del proyecto se implementa la actualización de la opinión de usuarios con base en la selección de algunos relaciones de influencia entre ellos.

Entrada: $G = (V, E)$, $\alpha : V \to [0..1], \beta : E \to [0..1]$ y $A \subseteq E$.

Salida: α^A.

4. Segunda parte

En esta parte del proyecto se implementa la actualización de la opinión de usuarios con base en la selección de algunas relaciones de influencia entre ellos, como en la parte anterior, con la diferencia de que el conjunto de relaciones a ser tenidas en cuenta se representa con una máscara de bits. Para ello, es necesario establecer un orden $\sqsubset$ (total) entre los arcos E del grafo G. Para cualesquiera (u_1, v_1) y (u_2, v_2) arcos en E, se tiene

$$(u_1, v_1) \sqsubset (u_2, v_2) \iff u_1 < u_2 \lor (u_1 = u_2 \land v_1 < v_2),$$

en donde $<$ es el orden usual sobre los números enteros. Este orden permite enumerar unívocamente los arcos en E con base en la secuencia 0, 1, ..., $|E| - 1$.

Una máscara de bits b de longitud $|E|$ representa un subconjunto de arcos de E de la siguiente manera: el bit menos significativo indica si el arco en la posición 0 (c.r.a. el orden $\sqsubset$) se tiene en cuenta (valor 1) o si no (valor 0), el segundo bit menos significativo indica si el arco en la posición 1 (c.r.a. el orden $\sqsubset$) se tiene en cuenta (valor 1) o si no (valor 0), el segundo bit menos, etc.

Por ejemplo, suponga que $E = \{(1, 3), (0, 3), (2, 1), (1, 2)\}$. Bajo el orden $\sqsubset$, se tiene

$$(0, 3) \sqsubset (1, 2) \sqsubset (1, 3) \sqsubset (2, 1).$$

Además, la máscara 1001 indica que se seleccionan los arcos $(0, 3)$ y $(2, 1)$.

Con base en lo anterior, se debe definir una función que resuelva el siguiente problema algorítmico.

Entrada: $G = (V, E)$, $\alpha : V \to [0..1], \beta : E \to [0..1]$ y $b \in \{0, 1\}^*$, con $|b| = |E|$.

Salida: α^A, en donde A es el conjunto de arcos en E seleccionados por b.

5. Tercera parte

En esta parte del proyecto se implementa la actualización de la opinión de usuarios una cantidad dada de veces. El problema algorítmico a resolver es el siguiente:

Entrada: $G = (V, E)$, $\alpha : V \to [0..1]$, $\beta : E \to [0..1]$, $N \in \mathbb{N}$ y $S \in \mathbb{N}$.

Salida: α_N.

El parámetro N indica la cantidad de veces que se actualizará la opinión de los usuarios en G, mientras que S es una semilla para la generación de números aleatorios. A continuación se explica la forma en la cual se generan los conjuntos de arcos que se usarán en cada una de las N actualizaciones del modelo.

La generación de números aleatorios se hará con ayuda del paquete `random` en Python3. Se creará un generador `rand` de números aleatorios de la siguiente manera:

```
rand = Random(S)
```

El generador `rand` se usará para generar la secuencia de números

$$b_0, b_1, \ldots, b_{N-1},$$

cada uno correspondiente a una máscara de $|E|$ bits. Cada una de estas máscaras se genera con el llamado

```
rand.randrange(0, 1<<len(E))
```

en donde `len(E)` indica la cantidad de arcos en E.

La función α_n se define inductivamente de la siguiente manera, para $0 \leq n \leq N$:

$$\alpha_n = \begin{cases} \alpha & , n = 0, \\ \text{“actualización de } \alpha_{n-1} \text{ con } b_{n-1}\text{”} & , n \neq 0. \end{cases}$$