Análisis de Agentes Automatizados en Redes Sociales: Un Enfoque Basado en Teoría de Grafos

Carlos Lucero, Samantha Montero, Camilo Becerra
Teoría de Grafos
Matemáticas Aplicadas y Ciencias de la Computación
Escuela de Ingeniería, Ciencia y Tecnología
Universidad del Rosario

I. Introducción

El internet ha evolucionado hasta convertirse en un espacio donde la interacción entre usuarios define gran parte de su funcionamiento. Sin embargo, en los últimos años ha surgido la teoría del «internet muerto», la cual sugiere que la cantidad de usuarios reales ha disminuido, mientras que la presencia de bots ha aumentado significativamente. En este contexto, los grafos de interacción en redes sociales han surgido como una herramienta poderosa para modelar y analizar estos cambios, permitiendo la identificación de patrones de comportamiento.

Los grafos en redes sociales son representaciones gráficas donde los usuarios se modelan como vértices y las interacciones entre ellos como aristas. Estos grafos capturan la estructura y dinámica de las redes, lo que permite aplicar algoritmos para identificar diferencias entre usuarios humanos y cuentas automatizadas.

II. Descripción del Problema

En la época digital, la existencia de agentes automatizados (bots) en las redes sociales constituye un reto crucial, dado que su comportamiento puede alterar la percepción de la interacción en línea y poner en riesgo la integridad de la información divulgada. El desafío consiste en reconocer, basándose en grandes cantidades de información y en un contexto de interacciones, patrones y conductas que faciliten la distinción entre usuarios auténticos y agentes automatizados.

Además, los bots suelen trabajar en "granjas" coordinadas, creando subgrafos con características estadísticas inusuales (por ejemplo, gran densidad interna pero aislamiento estructural en comparación con usuarios auténticos). Esto requiere una perspectiva multidimensional que incorpore la teoría de grafos, el aprendizaje automático y el análisis temporal para diferenciar entre actividad orgánica y automatizada, particularmente en situaciones donde ambas formas conviven.

III. Objetivos

III-A. Objetivo General

Desarrollar un modelo basado en teoría de grafos para la detección de bots en redes sociales, analizando la estructura de interacciones y aplicando algoritmos de detección de anomalías en la red.

III-B. Objetivos Específicos

- Modelar la red social como un grafo dirigido, donde los vértices representan usuarios y las aristas sus interacciones (seguimientos, menciones, retweets, etc.).
- Identificar patrones estructurales característicos de bots mediante métricas de teoría de grafos, como centralidad de grado, intermediación y coeficiente de agrupamiento.
- Implementar algoritmos de detección de comunidades (Louvain o Girvan-Newman) para identificar posibles "granjas de bots".
- Aplicar algoritmos de detección de anomalías como PageRank, detección de vértices con alta actividad y baja reciprocidad, y distribución de grados en la red.

IV. Marco Teórico

IV-A. Métricas estructurales

Para entender cómo funcionan las redes sociales, podemos imaginarlas como un dibujo hecho con puntos y líneas. Cada punto (llamado vértice) representa una cuenta de usuario (una persona o un bot), y cada línea (llamada arista) representa una conexión o interacción entre dos cuentas, como un "me gusta", una mención o un retweet.

Las siguientes herramientas matemáticas nos ayudan a entender el comportamiento de estas cuentas y a descubrir si son bots:

- Centralidad de grado (entrada/salida): cuenta cuántas conexiones tiene un usuario. Si una cuenta sigue a muchos pero nadie la sigue, eso es sospechoso.
- Centralidad de intermediación: muestra si una cuenta sirve como puente entre otras. Los bots normalmente no conectan grupos diferentes, por eso su puntuación aquí es baja.
- Coeficiente de agrupamiento: mide si los amigos de una cuenta también son amigos entre sí. Los humanos suelen formar grupos cerrados (como amigos del colegio), pero los bots no.
- PageRank: indica qué tan importante es una cuenta, según quién la sigue. Un bot puede seguir a muchos, pero si nadie importante lo sigue, no tiene buena reputación.

IV-B. *Grafos dinámicos*

Un grafo dinámico es una estructura que representa una red cuyos vértices y aristas pueden cambiar a lo largo del tiempo. A diferencia de los grafos estáticos, estos permiten modelar sistemas en evolución, como redes sociales o flujos de información; es decir, sus vértices y/o aristas pueden aparecer, desaparecer o cambiar sus propiedades conforme avanza el tiempo. Su función es capturar cómo varían las conexiones entre entidades, facilitando el análisis de patrones, cambios estructurales y comportamientos anómalos.

IV-C. *Algorítmo PageRank*

Este algoritmo ayuda a saber qué tan importante es cada usuario en la red. Imagina que todos los usuarios reparten un poco de su atención (como si fueran votos) a quienes siguen o mencionan. Las cuentas que reciben atención de otros importantes son aún más importantes.

Los bots suelen recibir poca atención verdadera, por eso su PageRank es bajo. De hecho, este algoritmo ya se usaba en Google para decidir qué páginas mostrar primero. Ejemplo de su implementación en Python:

```
import networkx as nx
G = nx.DiGraph()
# Anadir conexiones entre cuentas
pagerank_scores = nx.pagerank(G)
```

IV-D. *Algoritmo de Louvain*

Este algoritmo nos ayuda a encontrar grupos de cuentas que interactúan mucho entre sí, como si fueran un grupo de amigos o un club. Si encontramos un grupo muy cerrado que casi no habla con nadie más y todos hacen lo mismo, podría ser un grupo de bots.

El algoritmo va probando diferentes formas de agrupar las cuentas para encontrar la organización que mejor explique cómo se conectan entre ellas. Ejemplo de su implementación en Python:

```
import community
partition = community.best_partition(G)
```

IV-E. *Cálculo de métricas adicionales*

Aquí calculamos varios valores importantes para cada cuenta. Esto nos permite saber qué tan conectada está, si es un puente entre grupos y si está en un grupo unido. Ejemplo de su implementación en Python:

```
out_degrees = dict(G.out_degree())
in_degrees = dict(G.in_degree())
betweenness = nx.betweenness_centrality(G)
clustering = nx.clustering(G.to_undirected())
```

IV-F. *Reglas heurísticas de clasificación*

Después de tener todos los datos de cada cuenta, usamos reglas simples (como si fueran señales de alerta) para saber si una cuenta es un bot.

Por ejemplo, si alguien sigue a muchísimos, pero nadie lo sigue, publica demasiado, no tiene conexiones entre sus seguidores y nadie lo considera importante, probablemente no es una persona real.

Estas reglas no son perfectas, pero nos ayudan mucho para hacer una primera clasificación. Más adelante podríamos usar inteligencia artificial para mejorar esto. Función sencilla para etiquetar bots implementada en Python:

```
def es_sospechoso(vertice):
    return (
        out_degrees[vertice] > 100 and
        pagerank_scores[vertice] < 0.01 and
        clustering[vertice] < 0.1 and
        betweenness[vertice] < 0.001
    )
```

V. Modelamiento del Problema

Para detectar bots en redes sociales desde la perspectiva de teoría de grafos, es necesario traducir el comportamiento de los usuarios en estructuras formales que permitan análisis cuantitativos. Se parte del principio de que una red social puede representarse como un grafo dirigido $G = (V, E)$, donde V es el conjunto de vértices (usuarios) y E es el conjunto de aristas (interacciones), como menciones, retweets o seguidores. Esta representación captura no solo las relaciones explícitas entre cuentas, sino también su estructura global.

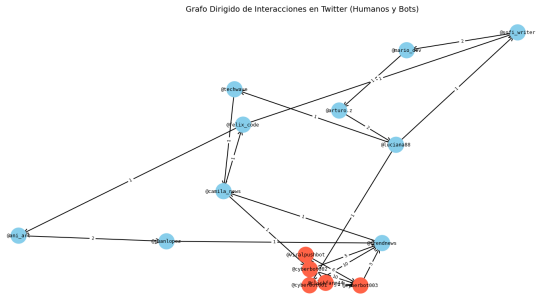

Figura 1. Representación básica de una comunidad y sus interacciones, inducidas en parte por ciertos agentes automatizados.

En la figura 1 muestra cómo interactúan cuentas humanas y bots en Twitter. Los vértices celestes representan humanos y los rojos a los bots. Se ve claramente que los bots están muy conectados entre sí, interactuando muchas veces, mientras que también se relacionan con algunas cuentas humanas, como `@trendnews` o `@camila_news`. Estas interacciones podrían indicar intentos de los bots por amplificar mensajes o influir en conversaciones. Por otro lado, los humanos tienden a estar en grupos más dispersos, pero algunos son puntos clave en la red. En general, el grafo deja ver una separación entre humanos y bots, aunque con ciertos vínculos entre ambos.

V-A. *Tipo de grafo y propiedades*

Se utilizará un grafo dirigido y ponderado, en el cual:

- Una arista dirigida $(u, v) \in E$ indica que el usuario u interactuó con el usuario v.
- El peso de la arista puede representar la frecuencia o el tipo de interacción (por ejemplo, retweets repetidos o menciones múltiples).

La red resultante suele ser dispareja (no todos los vértices tienen el mismo grado), con características típicas de redes reales como:

- Distribución de grado altamente sesgada (ley de potencias).
- Presencia de comunidades densamente conectadas.
- Escasa reciprocidad entre ciertos vértices (común en bots).

V-B. *Hipótesis estructurales para detección de bots*

Los bots presentan patrones estructurales particulares que los diferencian de usuarios humanos. Se modelan las siguientes hipótesis:

1. Los bots tienden a formar clústeres artificiales: subgrafos con alta densidad interna pero baja conectividad hacia el resto del grafo.
2. Las cuentas automatizadas muestran grados de salida altos y grados de entrada bajos.
3. El coeficiente de agrupamiento local es reducido: los bots no generan triángulos ni relaciones transversales.
4. El PageRank de un bot es bajo a pesar de su alta actividad, debido a su escasa importancia estructural.
5. La centralidad de intermediación es reducida, pues no suelen actuar como puentes entre comunidades.

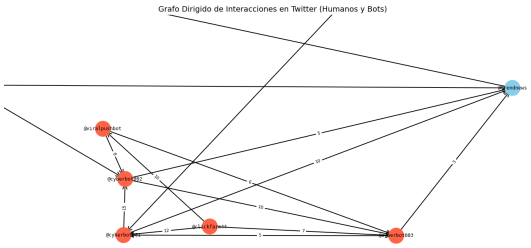

Figura 2. Representación de interacciones entre los agentes automatizados (clústeres artificiales) y su conexión con un usuario real.

V-C. *Representación dinámica y temporal*

Si se dispone de datos a lo largo del tiempo, se puede extender el modelado a grafos dinámicos $G_t = (V_t, E_t)$, donde los vértices y aristas pueden cambiar con el tiempo. Esto permite observar patrones sospechosos como:

- Crecimiento repentino del grado (seguidores artificiales).
- Aparición sincronizada de múltiples cuentas.
- Actividad intensiva durante ventanas temporales específicas.

VI. Solución Propuesta

Nuestra solución tiene varios pasos, que juntos nos ayudan a encontrar a los bots en la red:

1. Recolectar datos y construir el grafo:
   Tomamos información de una red social (real o simulada) y construimos el grafo. Cada cuenta es un vértice, y cada interacción es una flecha.
2. Calcular métricas:
   Medimos cosas como:
   - Cuántas conexiones tiene una cuenta.
   - Si está en medio de muchas rutas (como un puente).
   - Si está conectada a usuarios importantes.
   - Si sus amigos también se conocen entre sí.
3. Detectar comunidades:
   Usamos el algoritmo Louvain para descubrir grupos de cuentas que interactúan mucho entre sí. Si encontramos un grupo muy unido pero casi aislado, puede ser una 'granja de bots'.
4. Clasificar las cuentas:
   Finalmente, aplicamos reglas que nos dicen si una cuenta parece ser un bot. Por ejemplo:
   - Tiene muchas conexiones salientes.
   - Nadie importante la sigue.
   - Publica muy seguido.
   - No está bien conectada con otras cuentas humanas.

Si una cuenta cumple varias de estas condiciones, la marcamos como sospechosa.

VII. Bibliografía

Referencias

[1] R. Takacs and I. McCulloh, "Dormant Bots in Social Media: Twitter and the 2018 U.S. Senate Election," in *2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pp. 796-800, 2019.

[2] L. Amado Lapena, "Análisis del algoritmo PageRank: fundamento algebraico del orden de las búsquedas de Google," in *Universidad Politécnica de Madrid*, 2011. [En línea]. disponible en: https://oa.upm.es/69141/8/TFG_LAURA_AMADO_LAPENA.pdf. [Accedido: Febrero 2025].

[3] BaityBait, "El Internet que conocíamos SE HA ACABADO," in *YouTube*, [En línea]. disponible en: https://youtu.be/_Rim0DuxL7U?si=DocXX6uIMLH2VsZT, 2025. [Accedido: Febrero 2025].

[4] A.-L. Barabási, *Network Science*. Cambridge University Press, 2016.

[5] V. D. Blondel, J. L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *J. Stat. Mech.: Theory Exp.*, vol. 2008, no. 10, 2008.

[6] S. Cresci, R. Di Pietro, M. Petrocchi, A. Spognardi, and M. Tesconi, "The paradigm-shift of social spambots: Evidence, theories, and tools for the arms race," in *Proc. 26th Int. Conf. World Wide Web Companion*, pp. 963–972, 2017.

[7] D. Easley and J. Kleinberg, *Networks, Crowds, and Markets: Reasoning about a Highly Connected World*. Cambridge University Press, 2010.

[8] E. Ferrara, O. Varol, C. Davis, F. Menczer, and A. Flammini, "The rise of social bots," *Commun. ACM*, vol. 59, no. 7, pp. 96–104, 2016.

[9] S. Fortunato, "Community detection in graphs," *Phys. Rep.*, vol. 486, no. 3–5, pp. 75–174, 2010.

[10] M. E. J. Newman, *Networks: An Introduction*. Oxford University Press, 2010.

[11] V. S. Subrahmanian et al., "The DARPA Twitter bot challenge," *Computer*, vol. 49, no. 6, pp. 38–46, 2016.

[12] D. B. West, *Introduction to Graph Theory*, 2nd ed. Prentice Hall, 2001.

[13] A. Almaatouq, L. Radaelli, A. Pentland, and E. Shmueli, "Behavioral dynamics in social networks: Bot detection and beyond," *Sci. Adv.*, vol. 6, no. 17, 2020.

[14] Bot Repository – Indiana University, “Botometer: Datasets.” [En línea]. Available: https://botometer.osome.iu.edu/bot-repository/datasets.html [Accedido: Abril 2025].
[15] NetworkX Developers, “NetworkX documentation.” [En línea]. Available: https://networkx.org/documentation/stable/ [Accedido: Abril 2025].