Recomendación de usuarios basada en la topología de la red social

Joan Sol Roo[1], Marcio Caraballo Sosa[1]

[1]Universidad Nacional del Centro de la Provincia de Buenos Aires, Campus Universitario, Argentina

{jroo, mcaraballososa}@alumnos.exa.unicen.edu.ar

Resumen. Encontrar fuentes de información de buena calidad dentro de la comunidad de micro-blogging usando Twitter se vuelve esencial para los usuarios que utilizan esta plataforma en búsqueda de información; pero el enfoque tradicional de recomendación de usuarios basado en el análisis de contenido se dificulta debido al número de usuarios candidatos a evaluar en la red, en conjunto con limitaciones impuestas por Twitter para acceder a sus datos. En este trabajo se evalúa una alternativa de recomendación de usuarios basada únicamente en el estudio de la topología de red y en el estudio de usuarios similares.

Keywords: Sistemas de Recomendación de Información, Redes Sociales, Filtrado colaborativo, Twitter.

1 Introducción

Twitter es una red social con servicio de micro-blogging que permite a sus usuarios enviar y leer mensajes cortos de una longitud de 140 caracteres, llamados *tweets*. Dichos mensajes pueden tener cualquier contenido, sin embargo existen algunos usuarios particulares que publican tweets sobre un tema específico. Nos referiremos a estos usuarios que tratan temas específicos como *fuentes de información*, o simplemente, *fuentes*.

Esta red social permite *seguir* a usuarios, una forma de recibir actualizaciones constantes sobre las novedades publicadas. Así, cada usuario sigue a otros que son de su interés.

Las relaciones seguidor-seguido pueden ser simétricas, lo que puede considerarse una relación de *amistad*. Según estudios, estas relaciones son poco comunes en Twitter [1][2][3]. En contraste, muchos usuarios que utilizan Twitter (alrededor del 67.6%) son *buscadores de información* sobre sus temas de interés, siguiendo las fuentes de información que hablan de éstos. Es por esto que los usuarios de Twitter se beneficiarían con un sistema de recomendación de dichas fuentes de información.

Los sistemas de recomendación son sistemas específicos de filtrado de información que buscan recomendar ítems de interés al usuario. Para ello, extraen del perfil del usuario algunas características de referencia, en busca de predecir el ranking que él daría a un ítem que aún no ha evaluado.

En nuestro caso, utilizaremos las fuentes de información actualmente seguidas por un usuario como punto de partida, para obtener y luego ordenar las fuentes seguidas por *usuarios similares* (éstos son aquellos que poseen el mayor número de fuentes en común con el interesado).

Consideramos parte del perfil de un usuario de Twitter a los siguientes componentes, como:

- sus características básicas que permiten la *identificación* (nombre e ID de usuario);
- sus *relaciones* (seguidos y seguidores);
- su *contenido* (*tweets*);
- las *estadísticas* derivadas (cantidad de tweets, de seguidos y de seguidores).

Las anteriores representan al usuario dentro de todo Twitter (espacio de búsqueda); existe otro factor, atado a la búsqueda local realizada (subred): su *peso*, dado por la cantidad de *ocurrencias* al recorrer la red.

1.2 Limitaciones

Twitter provee una API, que permite generar aplicaciones que interactúen directamente con sus datos. La identificación del usuario dentro del sistema se realiza mediante el número de ID; para obtener el usuario completo (según definimos en el inciso anterior) hacen falta tres consultas a la API:

1. obtener sus características (nombre, y estadísticas);
2. obtener sus relaciones (seguidos y seguidores);
3. obtener su contenido (tweets).

La funcionalidad ofrecida se encuentra limitada tanto en cantidad de consultas (350 por hora) como en tiempo[1]. Si bien la primera limitación puede evadirse (utilizando múltiples cuentas), la segunda ofrece una gran restricción (dado que el uso de múltiples cuentas puede realizarse únicamente en serie, no en paralelo).

A fines de estudio, es posible generar un dataset, recolectando todos los datos de una subred a lo largo de un periodo de tiempo, para su posterior estudio; esto permite evaluar el funcionamiento teórico, pero no permite una aplicación en el entorno real.

2 Trabajos Relacionados

Existen múltiples estudios que fundan las bases para este trabajo: Hong et al.[4] estudian las *temáticas* y *categorías de temáticas* dentro de Twitter, mientras que Java

[1] Pruebas en distintos equipos con distintas configuraciones demoran en promedio 1 segundo en responder cada consulta; atribuimos entonces este tiempo al tiempo de respuesta de Twitter.

et al.[1] estudian las comunidades implícitas que existen en Twitter, que tratan temáticas relacionadas; dentro de estas comunidades los usuarios pueden categorizarse en: *fuentes*, *buscadores de información*, o bien *amigos*.

Otros autores han encontrado que hay información útil en la topología de la red por sí misma: Chen et al. [5] compararon algoritmos basados en relaciones y en contenido para recomendaciones de usuarios, encontrando que el primero es mejor detectando contactos conocidos mientras que el segundo en encontrar nuevas relaciones.

Por otro lado existen trabajos que abordan el problema de recomendación de usuarios dentro de Twitter: Sun et al. [6] utilizaron un algoritmo basado en difusión para obtener un grupo de usuarios que toman el rol de reporteros en emergencias; más relacionado a nuestro trabajo, están los algoritmos de recomendación de usuarios en Twitter a partir de un subconjunto de usuarios presentado por Hannon et. al [7]. Estos autores consideraron múltiples estrategias de generación de perfiles de acuerdo a cómo los usuarios son representados en un enfoque basado en contenidos (sus tweets y los de los usuarios con los que se relacionan), un enfoque basado el filtrado colaborativo (basados en los IDs de sus seguidores y/o de sus seguidos), así como enfoques híbridos.

Este trabajo es un caso particular de filtrado colaborativo, que explota las características de las comunidades, en particular, temáticas e intereses en común; mediante el estudio de las relaciones actuales, se da lugar a un perfil de usuario[9], a mayor cantidad de fuentes siga el usuario, mejores serán los resultados obtenidos.

3 Recomendación de Usuarios basada en Topología

El objetivo de este trabajo es obtener resultados aceptables en el menor tiempo posible, explotando la semántica que conllevan las relaciones entre usuarios, basándonos en dos premisas:

- las fuentes que sigue un usuario potencialmente lo ubican dentro de comunidades;
- dados dos usuarios que siguen fuentes de información, se pueden considerar más o menos similares acorde a la cantidad de fuentes en común. Si dos usuarios son suficientemente similares, las fuentes de uno de ellos pueden sugerirse al otro.

El algoritmo planteado (Algoritmo 1) explota las características expuestas en el punto anterior propagando su peso (número de ocurrencias) a lo largo de la red. En cada paso del recorrido, el peso de un usuario es la suma de los pesos de los usuarios del nivel anterior con los que se relaciona; se ordenan por la cantidad de ocurrencias y luego se conservan los ***n*** primeros (*tamaño máximo*).

Algoritmo1 – recomendación basada en topología

Entrada: usuario interesado
Salida: vector de usuarios a recomendar

```
01:  Vector<Usuario> recomendar(Usuario inicial)
```

```
02: inicial.ocurrencias = 1
03: Vector<Usuario> resultado = {inicial}
04: por cada (Nivel nivel en recorrido)
05:   resultado = nivel.obtenerDesde(resultado)
06: devolver resultado
```

```
07: Vector<Usuario>
    Nivel.obtenerDesde(Vector<Usuario>  actuales)
08:Vector<Usuario> resultado
09:para cada(Usuario actual en actuales)
10: si(!visitado(actual))
11:  visitado(actual)=true;
12:  para cada(Usuario siguiente en siguientes(actual))
13:    si(!visitado(siguiente))
14:       si(!resultado.contiene(siguiente))
15:         resultado.agrega(siguiente)
16:         siguiente.ocurrencias = actual.ocurrencias
17:       sino
18:         siguiente.ocurrencias += actual.ocurrencias
19:para cada (Usuario usuario en resultado)
20:  si(!criterio.cumple(usuario))
21:    resultado.quitar(usuario)
22:resultado.ordenar(porOcurrencias)
23:mientras(resultado.size()>tamaño_maximo)
24:  resultado.descartarUltimo
25:devolver resultado;
```

En este algoritmo, se obtiene información únicamente de los ***n*** usuarios que son preseleccionados. Es entonces posible evaluar una topología con cientos de miles de usuarios por nivel, en tiempos razonables, realizando consultas únicamente sobre los más relevantes.

El pseudocódigo indicado es genérico a cualquier recorrido de la red social. Cada nivel puede ser de seguidos o seguidores (línea 13: la función siguientes(Usuario) depende del tipo de nivel).

Así también, el criterio de descarte de usuarios (línea 20) depende del filtrado deseado. En particular, las fuentes de información se destacan por tener mayor cantidad de seguidores que seguidos; para determinar si un usuario es una potencial fuente de información, se esperaba que:

Si U *Fuente* entonces *U.seguidores* > *2*U.seguidos*

Es decir que consideramos fuentes de información sólo a aquellos usuarios que son seguidos por al menos el doble de usuarios que ellos mismos siguen. Los niveles utilizados en la recomendación de fuentes se muestran en la Tabla 1.

Tabla 1. Niveles de recorrido para la recomendación de fuentes.

	Tipo	Criterio	tamaño máximo (n)	Resultado
Nivel 1	Seguidos	Fuente	200	Fuentes actuales
Nivel 2	Seguidores	-	200	Usuarios similares
Nivel 3	Seguidos	-	20	Fuentes a recomendar

A continuación, en la figura 1 se exponen los tipos de usuarios y relaciones que se consideran en el recorrido de la red planteado.

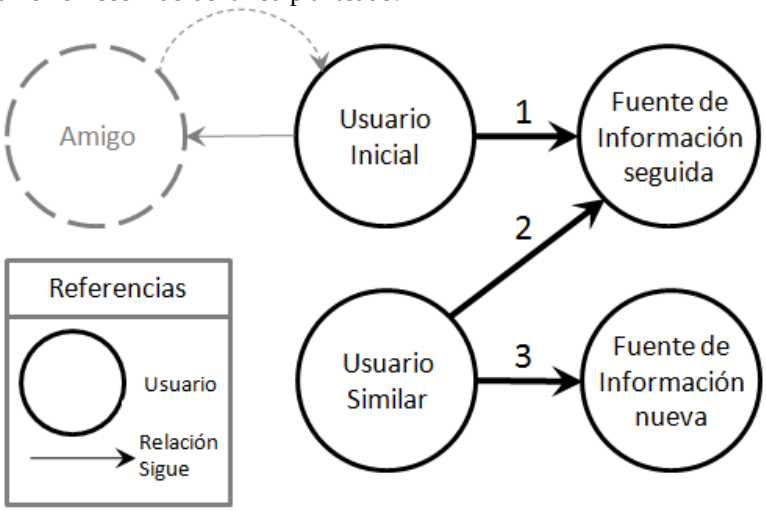

Fig. 1. Recorrido de la red para recomendación de fuentes a partir de *fuentes seguidas*.

4 Evaluación Experimental

Para la evaluación, se realizaron experimentos sobre 307 usuarios obtenidos al azar de entre el total de usuarios actualmente existentes en Twitter, tomando como única restricción que su número de seguidos se encuentre entre 50 y 350; el límite inferior permite obtener una muestra válida para comparar contra los algoritmos de contenido[8], y el límite superior evita procesar redes demasiado extensas.

Los resultados obtenidos serán evaluados con dos métricas:

1. Precisión en K (P@K): porcentaje de correctos entre los primeros K recomendados.
2. Recuperación o *Recall*: porcentaje de recuperados de entre el total de correctos.

4.1 Evaluación mediante Redescubrimiento de Red (Hold Out)

En este experimento se utilizó la técnica de *hold out*, ocultando el 30% de las fuentes originalmente seguidas por el usuario al cual se desea recomendar y se evaluando qué porcentaje de éstas fueron recuperados por el algoritmo. También se calculó cuáles de estos usuarios fueron redescubiertos dentro de las primeras 20 recomendaciones. Para ello, sólo se evaluarán los usuarios con al menos 40 seguidos, por dos razones: es el valor utilizado por el algoritmo contra el que se compararán los resultados, y, así también, da lugar a que se recuperen suficientes resultados como para que la precisión de 1 a 20 tenga peso.

El valor de recuperación fue de 97,59%. El ordenamiento de los resultados devuelve, en promedio, el total de los recuperados dentro del primer 4,84% de los usuarios evaluados, reduciendo entonces el conjunto de búsqueda de manera drástica. En cuanto a la distribución de probabilidad[2] de recuperación de usuarios esperados, se evaluó el valor promedio de *posición mínima* (primer encontrado), *mediana* (donde se han obtenido la mitad de los recuperados) y *máxima* (la posición del último encontrado), la figura 2 refleja estos valores.

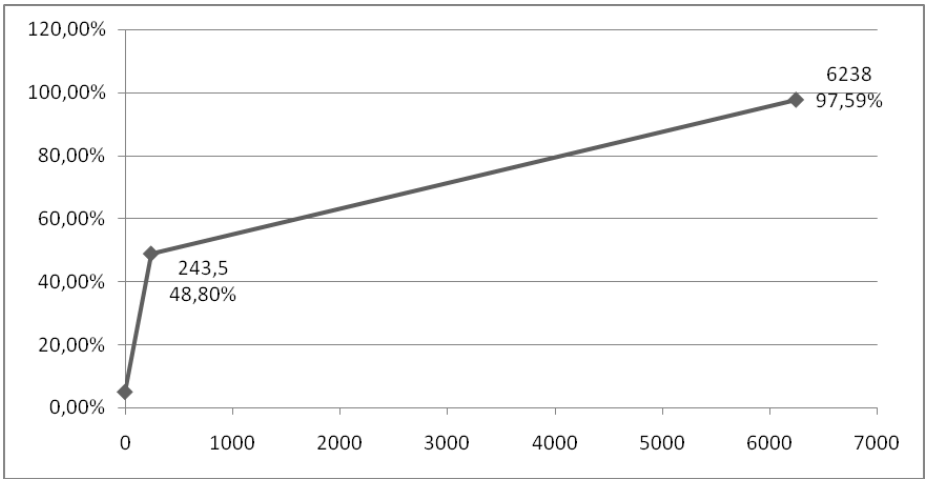

Fig. 2. Porcentaje recuperado vs posición.

Como se puede observar en la figura 2, la distribución de probabilidad se asemeja a una distribución exponencial, recuperando la mitad de los resultados esperados en aproximadamente primer 10,94%. Esto implica que es más probable obtener cada usuario deseado entre los primeros recomendados, que entre los últimos; en particular, dentro del intervalo en el que se evalúa la precisión; esto parece implicar que la precisión se ve afectada por la cantidad de fuentes ocultas.

Para estudiar la precisión, compararemos los resultados con los obtenidos al recomendar usuarios usando *categorías de interés* [8]:

[2] La distribución de probabilidad de una variable aleatoria es una función que asigna a cada suceso definido sobre la variable aleatoria la probabilidad de que dicho suceso ocurra.