

**UNIVERSIDAD AUTÓNOMA DE MADRID
ESCUELA POLITÉCNICA SUPERIOR**



**Máster Universitario en Investigación e Innovación en Inteligencia Computacional y
Sistemas Interactivos**

TRABAJO FIN DE MÁSTER

**Evolución de redes sociales con recomendación
interactiva de contactos**

**Autor: Antonio Amor Mourelle
Tutor: Pablo Castells Azpilicueta**

agosto 2020

Todos los derechos reservados.

Queda prohibida, salvo excepción prevista en la Ley, cualquier forma de reproducción, distribución comunicación pública y transformación de esta obra sin contar con la autorización de los titulares de la propiedad intelectual.

La infracción de los derechos mencionados puede ser constitutiva de delito contra la propiedad intelectual (*arts. 270 y sgts. del Código Penal*).

DERECHOS RESERVADOS

© 22 de julio de 2020 por UNIVERSIDAD AUTÓNOMA DE MADRID

Francisco Tomás y Valiente, nº 1

Madrid, 28049

Spain

Antonio Amor Mourelle

Evolución de redes sociales con recomendación interactiva de contactos

Antonio Amor Mourelle

Lo fácil es hacer la fórmula difícil.

Pablo Castells

AGRADECIMIENTOS

Este año ha sido un año distinto y difícil para todos por culpa del COVID-19. En especial para todos aquellos que han perdido a alguien. Quiero utilizar estas líneas para acordarme de todos los que ya no están y mandarles a sus familias ánimos y mis condolencias.

RESUMEN

Cada vez estamos más acostumbrados a ver sistemas de recomendación. Esto se debe a que las grandes plataformas como Amazon, Netflix o Twitter las utilizan a diario para proveer contenido a sus usuarios y mejorar sus servicios. Sin embargo, los sistemas de recomendación están lejos de ser un problema cerrado. Este TFM estudia el impacto que tiene un sistema de recomendación de contactos dentro de una red social.

En primer lugar se observan diferentes características de la red y su evolución al permitir a un recomendador sugerir nuevos contactos dentro de la red. Además se analiza si existe algún algoritmo que destaque en comparación a los demás al estudiar cada característica. Este análisis se lleva a cabo mediante la implementación de un estudio en el que se compara el comportamiento obtenido por algoritmos clásicos en la recomendación, como son los algoritmos basados en vecindarios, con algoritmos presentes a día de hoy en el estado del arte: los bandidos multi-brazo. Los resultados muestran que dependiendo de que característica queramos maximizar debemos utilizar los algoritmos basados en vecindarios o los bandidos multi-brazo.

Además, el TFM ahonda en uno de los temas abiertos para los bandidos multi-brazo: considerar nuevas nociones de acierto. Habitualmente se considera acierto simplemente cuando el item recomendado esta en el conjunto de test (suponiendo un experimento offline). El TFM entra a considerar como acierto nuevas nociones como el aumento de alguna de las características de la red social, con el fin de observar si a largo plazo el bandido multi-brazo es capaz de descubrir como maximizar dicha característica. Los resultados resultan prometedores, de los casos estudiados se observa como uno de los bandidos multi-brazo mejora el resultado de la característica medida en todos los casos.

PALABRAS CLAVE

Sistema de recomendación, algoritmos basados en vecindarios, precisión, contacto, recomendación de contactos, Bandidos Multi-brazo

ABSTRACT

We are increasingly used to seeing recommender systems. This is because large platforms such as Amazon, Netflix or Twitter use them on a daily basis to provide content to their users and improve their services. However, recommender systems are far from being a closed problem. This thesis studies the impact that a contact recommender system has within a social network.

Firstly, it looks at different characteristics of the network and its evolution by allowing a referrer to suggest new contacts within the network. In addition, we analyze if there is any algorithm that stands out in comparison to others when studying each feature. This analysis is carried out through the implementation of a study in which the behavior obtained by classic algorithms in the recommendation, such as neighborhood-based algorithms, is compared with algorithms present today in the state of the art: the multi-armed bandits. The results show that depending on which characteristic we want to maximize we should use the neighborhood-based algorithms or the multi-armed bandits.

In addition, the thesis delves into one of the open issues for multi-armed bandits: to consider new notions of success. Usually it is considered a hit simply when the recommended item is in the test suite (assuming an offline experiment). This paper considers new notions as success, such as the increase of some of the characteristics of the social network, in order to observe if in the long term the multi-armed bandit is able to discover how to maximize such characteristic. The results are promising, from the studied cases it is observed how one of the multiarmed bandits improves the result of the measured characteristic in all cases.

KEYWORDS

Recommender system, neighborhood-based algorithms, accuracy, contact, contact recommendation, Multi-Arm Bandits

ÍNDICE

1	Introducción	1
1.1	Motivación	1
1.2	Objetivos	1
1.3	Metodología	2
2	Estado del arte	3
2.1	Sistemas de recomendación	3
2.2	Recomendación clásica	4
2.2.1	Filtrado colaborativo	4
2.2.2	Sistemas de recomendación basados en contenido	4
2.2.3	Sistemas de recomendación híbridos	5
2.3	Formalización de la recomendación clásica	5
2.3.1	Recomendación de contactos	6
2.4	Recomendación interactiva	7
2.4.1	Bandidos Multi-brazo	7
2.4.2	Formalización de los Bandidos Multi-brazo	8
3	Algoritmia	11
3.1	Algoritmos implementados	12
3.1.1	Algoritmos básicos	12
3.1.2	Algoritmos basados en vecindarios	13
3.2	Algoritmos de implementación externa: Bandidos Multi-brazo	15
3.2.1	ϵ -Greedy	15
3.2.2	Thompson Sampling	15
4	Evaluación y Métricas	17
4.1	Métricas basadas en distancia	18
4.1.1	Reciprocal Diameter	18
4.1.2	Reciprocal Average Eccentricity	19
4.2	Métricas de diversidad estructural	20
4.2.1	Degree Gini Complement	20
4.2.2	Métricas locales en diversidad	20
4.2.3	Métricas globales en diversidad	22
4.3	Métricas de novedad	22

4.3.1	<i>Long Tail Novelty</i>	23
4.4	Métricas de diversidad	23
4.4.1	<i>Prediction Gini Complement</i>	24
5	Experimentos	25
5.1	Configuración del experimento	25
5.1.1	Datos utilizados	25
5.2	Comparativa de algoritmos	26
5.3	Nuevas funciones de recompensa en Bandidos Multi-brazo	32
6	Conclusiones	35
6.1	Resumen y contribuciones	35
6.2	Trabajo futuro	36
	Bibliografía	37

LISTAS

Lista de ecuaciones

2.1	Enfoque probabilístico del problema de recomendación	6
2.2	Enfoque de <i>ranking</i> del problema de recomendación	6
2.3	Formulación general del problema de recomendación por <i>rankings</i>	6
2.4	Formulación por <i>ranking</i> para la recomendación de contactos	6
2.5	Recompensa acumulada de un <i>item</i> i hasta tiempo t	8
3.1	Fórmula para la recomendación aleatoria	12
3.2	Fórmula para la recomendación por popularidad	12
3.3	Fórmula de MCN	13
3.4	Fórmula de Jaccard	14
3.5	Fórmula de Adamic	14
3.6	Fórmula de Resource Allocation	14
3.7	Fórmula de $\epsilon - Greedy$	15
3.8	Fórmula de Thompson Sampling	15
4.1	Definición de Diámetro de un grafo G	18
4.2	Definición de <i>Reciprocal Diameter</i>	19
4.3	Definición de excentricidad de u	19
4.4	Definición de <i>Reciprocal Average Eccentricity</i>	19
4.5	Definición de <i>Degree Gini Complement</i>	20
4.6	Definición de <i>Embeddedness</i>	21
4.7	Definición de <i>Average Edge Weakness</i>	21
4.8	Definición de <i>Clustering Coeficient Complement</i>	21
4.9	Definición de <i>Community Edge Gini Complement</i>	22
4.10	Definición de <i>Long Tail Novelty</i>	23
4.11	Definición de <i>Prediction Gini Complement</i>	24

Lista de figuras

2.1	Ejemplo gráfico de recomendación por filtrado colaborativo	4
2.2	Ejemplo gráfico de recomendación basada en contenido	5
3.1	Algoritmos de recomendación de contactos tratados en el TFM	11

3.2	Posibles vecindarios para un usuario	13
4.1	Ejemplo distancia $\delta(u, v)$	18
4.2	Ejemplo diámetro de un grafo G	19
4.3	Ejemplo de enlace débil	21
5.1	Comparativa de <i>Reciprocal Average Eccentricity</i> para los diferentes algoritmos de recomendación	27
5.2	Comparativa de <i>Degree Gini Complement</i> para los diferentes algoritmos de recomendación	28
5.3	Comparativa de <i>Average Edge Weakness</i> para los diferentes algoritmos de recomendación	29
5.4	Comparativa de <i>Clustering Coeficient Complement</i> para los diferentes algoritmos de recomendación	30
5.5	Comparativa de <i>Comunity Edge Gini Complement</i> para los diferentes algoritmos de recomendación	30
5.6	Comparativa de <i>Long Tail Novelty</i> para los diferentes algoritmos de recomendación ..	31
5.7	Comparativa de <i>Prediction Gini Complement</i> para los diferentes algoritmos de recomendación	32
5.8	Resultados de utilizar DGC como acierto en los bandidos multi-brazo	33
5.9	Resultados de utilizar CCC como acierto en los bandidos multi-brazo	34
5.10	Resultados de utilizar CGEC como acierto en los bandidos multi-brazo	34

Lista de tablas

2.1	Correspondencia entre sistemas de recomendación y bandidos multi-brazo	9
5.1	Detalles de la red generada por los datos	26
5.2	Mejores configuraciones para los algoritmos basados en vecindarios para la recomendación estática	26

INTRODUCCIÓN

Es este capítulo se detalla la introducción del TFM. Empezaremos exponiendo la motivación por la que se realiza este proyecto y a continuación se muestran los objetivos marcados y la metodología utilizada.

1.1. Motivación

En la actualidad las redes sociales son uno de los sistemas mas utilizados por el grueso de la población. Facebook, Instagram y Twitter son quizás los ejemplos más significativos de redes sociales pero existen infinidad más: LinkedIn, que se ha logrado un hueco en el ámbito laboral, google+, que fue un intento fallido más de Google de entrar en este sector o Pinterest, entre muchas otras. A día de hoy, todas las redes sociales implementan sistemas de recomendación. En Instagram, toda una sección basada en tus contactos y en tus búsquedas. LinkedIn por su parte te recomienda también *post* de los usuarios y empleos que anuncian las empresas en los que podrías encajar. Pero además, en lo que coinciden todas las redes sociales es en la recomendación de contactos dentro de la misma. Esto es natural puesto que a la plataforma le interesa que conozcas gente con la que nutrirte de contenido la red y así seguir ofreciéndote una experiencia de usuario satisfactoria.

El presente TFM se centra en el impacto que tiene un sistema de recomendación de contactos sobre la red social, desde un punto de vista estructural de la red, analizando las variaciones del grafo y sus propiedades cuando se realizan recomendaciones. Pero además lo hace desde una novedosa perspectiva interactiva de las recomendaciones en la que la red crece por las recomendaciones realizadas, pero además las recomendaciones del futuro dependen de la aceptación o no por parte del usuario de la recomendaciones que se le han realizado hasta el momento.

1.2. Objetivos

La motivación anterior nos lleva a plantearnos como principal objetivo el estudio de la evolución de la red social cuando esta tiene por detrás un sistema de recomendación de contactos que la hace evolu-

cionar de manera fluida. Concretamente se plantea observar la evolución del grafo cuando utilizamos diferentes algoritmos de recomendación. Esto puede dividirse en las siguientes cuestiones:

- Observar la evolución de diferentes métricas cuando introducimos la perspectiva interactiva en algoritmos clásicos de recomendación de contactos.
- Utilizar algoritmos de recomendación por refuerzo (que llevan intrínseco el carácter cíclico) y compararlos con los algoritmos clásicos.
- Modificar el funcionamiento de los algoritmos de recomendación por refuerzo para que consideren otras métricas más allá del acierto.

1.3. Metodología

Para lograr los objetivos descritos, se sigue el siguiente procedimiento de trabajo. En primera instancia, se ha estudiado el estado del arte en lo relacionado a la tarea de recomendación, centrándose en los matices que ofrece la recomendación de contactos. Al mismo tiempo se ha consultado documentación sobre la recomendación interactiva.

Tras entender el problema se ha implementado en Python un módulo de recomendación de contactos, al que se le ha añadido la perspectiva interactiva, con el fin de poder observar la evolución de la red tras realizar varias recomendaciones. A este módulo le acompañan varios algoritmos de recomendación de contactos. Además se han utilizado librerías externas para obtener las versiones habituales de los algoritmos de aprendizaje por refuerzo utilizados en este TFM. A dichos algoritmos se les ha realizado las modificaciones pertinentes con el fin de cumplir con el tercer de los objetivos expuestos en la sección anterior.

Por último se han diseñado y ejecutado una serie de experimentos. En primer lugar se han adaptado los algoritmos de recomendación clásicos a la perspectiva interactiva que se propone en este TFM y se ha observado el impacto que tienen estos algoritmos al utilizarlos como recomendadores en la red a largo plazo. Además, estos resultados se han comparado con los obtenidos por los bandidos multi-brazo, presentes en el estado del arte. El segundo experimento consiste en variar la noción de acierto en los bandidos multi-brazo, con el fin de investigar si estas pueden producir mejores resultados para objetivos más concretos más allá del acierto en la recomendación.

ESTADO DEL ARTE

Los sistemas de recomendación han sufrido un fuerte auge durante los últimos años, especialmente con la aparición de servicios de *streaming* y el auge del sector *retail*. Esto ha llevado a que se desarrollen diferentes técnicas para llevar a cabo la tarea de recomendación. En este capítulo se detallan dichas técnicas y ahonda en sus similitudes y sus diferencias, para así situar al lector en el momento actual en el que se encuentran los sistemas de recomendación. Además se realizarán las matizaciones pertinentes para centrar al lector en la tarea específica de la recomendación de contactos.

2.1. Sistemas de recomendación

Los sistemas de recomendación se han vuelto una herramienta indispensable en muchas de las grandes plataformas utilizadas por los usuarios a día de hoy. Ya no solo por el gran volumen de datos que estas manejan (Amazon tiene a disposición de sus clientes más de 400 millones de artículos) sino por que cada vez los usuarios están más acostumbrados a su uso dentro de estas plataformas (se han realizado estudios ¹ que estiman que el 75% de las visualizaciones de Netflix se producen tras una recomendación).

Habitualmente se han planteado los estudios de sistemas de recomendación como algo estático. Sin embargo, las grandes plataformas evolucionan a gran velocidad y tienen un catálogo a recomendar que sufre constantes cambios. Esto ha hecho que el enfoque estático se quede desfasado y ha dado lugar a técnicas de aprendizaje cíclico en los sistemas de recomendación. Este aprendizaje se realiza mediante la utilización del sistema por parte de los usuarios. Durante dicho uso el sistema realizará recomendaciones y en función de la aceptación (o no) de las mismas el sistema adapta las recomendaciones futuras. El objetivo final del sistema cambia al introducir este nuevo paradigma, ya no se busca la mayor aceptación en el momento de realizar la recomendación, sino que se plantea maximizar la satisfacción del usuario a largo plazo, incluso si esto supone pequeños fallos en la recomendación de carácter más inmediata.

¹ <https://www.mckinsey.com/industries/retail/our-insights/how-retailers-can-keep-up-with-consumers>

2.2. Recomendación clásica

Los sistemas de recomendación clásicos surgen del paradigma del aprendizaje supervisado. Toman datos de valoraciones que los usuarios de sistema han realizado sobre los *items* disponibles y el sistema debe aproximar la valoración que dichos usuarios realizarían si disfrutasen de contenido que todavía no han utilizado. Para realizar esta tarea en la práctica se acude a particiones *Train-Test*, que tienen un carácter estático. Para este tipo de recomendación, se pueden diferenciar 3 grandes grupos en función del enfoque seguido para realizar las recomendaciones.

- **Basados en filtrado colaborativo** [2, chap. 1.3.1], recomienda contenido que resultó satisfactorio para usuarios similares, para ello se utilizan las características de los usuarios, tanto explícitas (sexo, edad,...) como implícitas (gustos observados, clicks...).
- **Basados en contenido** [2, chap. 1.3.2], utiliza las características de los *item* para recomendar contenido similar al que a sido marcado como satisfactorio previamente por el usuario.
- **Híbridos** [3], combinaciones de los anteriores .

2.2.1. Filtrado colaborativo

Un sistema de recomendación se dice que es de filtrado colaborativo cuando utiliza los gustos de los usuarios para seleccionar que contenido va a recomendar. Concretamente observa las valoraciones que diferentes usuarios han realizado sobre *items* concretos y si son parecidas considera que el contenido que un usuario, u , ha visto puede ser de interés para usuarios parecidos como v . Este comportamiento se ilustra en la figura 2.1.

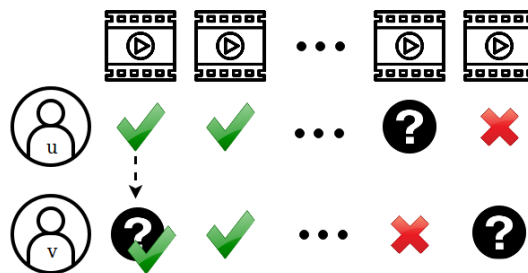


Figura 2.1: Ejemplo gráfico de recomendación por filtrado colaborativo

2.2.2. Sistemas de recomendación basados en contenido

Los sistemas de recomendación se consideran basados en contenido cuando estos solo utilizan las similitudes entre *items* para recomendar. Un ejemplo fácil de visualizar podría ser un nuevo usuario de Netflix, pongamos que ve una película de espías. Entonces el sistema le recomendará películas del

mismo tipo. Es el tipo de recomendación más intuitiva pero conlleva ciertos problemas, concretamente el encasillamiento del usuario en los temas concretos y la falta de descubrimiento de nuevo contenido de la plataforma. Un ejemplo de funcionamiento de un sistema de recomendación basado en contenido puede verse en la figura 2.2.

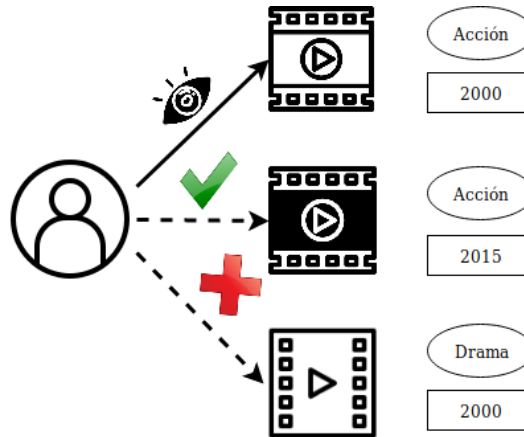


Figura 2.2: Ejemplo gráfico de recomendación basada en contenido

2.2.3. Sistemas de recomendación híbridos

Los sistemas híbridos surgen con el objetivo de aprovechar las ventajas de los algoritmos anteriores a la vez que se palían las desventajas de los mismo. A raíz de los sistemas de recomendación expuestos anteriormente aparecen 2 tipos de sistemas híbridos:

- **Combinación de los algoritmos**, se crea un único sistema de recomendación con un algoritmo que combina el filtrado colaborativo y la recomendación basada en contenido.
- **Mezcla de los resultados**, se emplean diferentes recomendadores cada una con una filosofía distinta y posteriormente se combinan los resultados que estos proporcionan.

2.3. Formalización de la recomendación clásica

En esta sección se va a formalizar la recomendación clásica tratada en la sección 2.2. Como se comentó anteriormente, recomendar se puede entender como asignarle a un *item* la valoración que un usuario tendría sobre dicho *item*. De esta definición ya salen los 3 primeros conceptos que debemos definir:

- \mathcal{U} , conjunto de los usuarios que utilizan el sistema de recomendación.
- I , conjunto de *items* que pueden ser recomendados por el sistema.
- I_u , subconjunto de I con los *items* con los que el usuario $u \in \mathcal{U}$ está satisfecho.

Con estos conceptos ya podemos definir la primera formulación de recomendar un *item* i a un usuario u como se observa en 2.1:

$$rec_u(i) = \begin{cases} i & \text{si } \mathcal{P}(i \in I_u) \geq x \\ \emptyset & \text{si } \mathcal{P}(i \in I_u) < x \end{cases} \quad (2.1)$$

donde $i \in I$, $u \in \mathcal{U}$ y $x \in [0, 1]$ una probabilidad limite a partir del cual se considera que i es del gusto de u .

Pero la definición anterior puede resultar poco práctica en situaciones reales donde se pretende siempre recomendar k *items*, ya que es posible que ninguna probabilidad super el umbral x . Es por ello que realizamos una modificación a la definición para deshacernos de este parámetro y pasar a recomendar k *items*.

$$rec_u = \{i_j \mid \max_k(\mathcal{P}(i_j \in I_u))\} \quad (2.2)$$

donde $i_j \in I$ y $u \in \mathcal{U}$.

Ahora que ha desaparecido el umbral, ya no necesitamos tener una cota para la función con la que recomendamos, por ello, para hacer más general el planteamiento se prescinde de la probabilidad \mathcal{P} y admitimos cualquier función f , tal y como se muestra en 2.3.

$$rec_u = \{i_j \mid \max_k(f_u(i_j))\} \quad (2.3)$$

donde $i_j \in I$ y $u \in \mathcal{U}$.

Los diferentes tipos de función f dan lugar a los diferentes algoritmos utilizados en la recomendación clásica detallados en la sección 2.2.

2.3.1. Recomendación de contactos

Este TFM se enfoca en la recomendación de contactos [4]. Este tipo concreto de recomendación viene principalmente por las redes sociales, y tiene como objetivo descubrir relaciones sociales ya existentes fuera de la red social e incentivar la creación de nuevas relaciones a través de la red social.

El problema de la recomendación de contactos se puede formular a partir de 2.3, pero en este caso concreto destaca que el conjunto de *items* pertenece al mismo espacio que el conjunto de usuarios. Por esto, la formula 2.3 se modifica como se muestra a continuación:

$$rec_u = \{v \mid \max_k(f_u(v))\} \quad (2.4)$$

donde $u, v \in \mathcal{U}$. Además la recomendación de contactos tiene un restricción que la vuelve especial con respecto a las demás recomendaciones, no es posible volver a recomendar un usuario v que ya

pertenece a los contactos de u . Por eso es necesario añadir la restricción: $v \notin I_u$.

2.4. Recomendación interactiva

La recomendación interactiva llevada a cabo en los últimos años por los grandes sistemas de recomendación supone un cambio en la forma de entender la recomendación. El objetivo sigue siendo realizar recomendaciones certeras y descubrir nuevo contenido de interés a los usuarios pero ahora se prioriza el obtener dichos objetivos en el medio-largo plazo y se entiende que para conseguir esto es posible que se obtengan peores resultados en un primer momento.

Este enfoque entra dentro de una de las áreas del Aprendizaje Automático, concretamente el Aprendizaje por Refuerzo. En el Aprendizaje por Refuerzo un agente interactúa con su entorno y a cada interacción le corresponde una recompensa. A partir de las recompensas actualiza su base de conocimiento y por tanto su patrón de comportamiento, todo con el objetivo de maximizar las ganancias en el largo plazo. Y para poder garantizar en cierta manera que las ganancias a largo plazo son óptimas el agente debe realizar un equilibrio entre acciones de exploración (del entorno) y acciones de explotación (del *item* que proporciona mayor recompensa). Las acciones de exploración se corresponden con aquellas que pretenden descubrir nuevo conocimiento. Un ejemplo sería recomendar una nueva película recién estrenada en Netflix. Si las primeras recomendaciones son satisfactorias para los usuarios el sistema reconoce que es una buena película y puede empezar a recomendarla al grueso de los usuarios de Netflix. A esta segunda fase se la conoce como la fase de explotación y se produce con el fin de maximizar la recompensa media obtenida por el sistema con el conocimiento actual que este posee.

Este TFM se centra en una de las ramas principales del Aprendizaje por Refuerzo concretamente en los Bandidos Multi-brazo.

2.4.1. Bandidos Multi-brazo

Los Bandidos Multi-brazo son un tipo de Aprendizaje por Refuerzo. Se caracterizan por que un agente debe elegir entre diferentes acciones posibles, las cuales no generan la misma recompensa cada vez que dicha acción es seleccionada. Lo que se asume es que cada acción tiene asociada una distribución para proporcionar una recompensa y el objetivo final del agente es balancear la exploración y la explotación de las acciones (llamadas brazos en la literatura) con el fin de encontrar aquellos brazos con la mayor recompensa media.

Existen diferentes tipos de Bandidos Multi-brazo y sus diferencias vienen de la forma que tienen de gestionar el balance entre exploración y explotación. En este TFM se tratan 2 de los Bandidos Multi-brazo más habituales, concretamente ϵ -Greedy y Thompson Sampling, los cuales se detallarán

posteriormente en la sección 3.2.

2.4.2. Formalización de los Bandidos Multi-brazo

Esta sección tiene como objetivo formalizar el uso de Bandidos Multi-brazo para la tarea de recomendación de contactos. Como se comentó en la sección 2.3 recomendar se entiende como asignarle la valoración que un usuario daría sobre un *item* cuando este lo disfrutase. Sin embargo, en la recomendación de contactos merece la pena realizar una pequeña variación: consideraremos que las valoraciones solo pueden ser binarias, siendo el 1 cuando el sistema considera que v es una buena recomendación para u y que se aceptará dicha recomendación y siendo 0 en caso contrario.

Siguiendo con la formalización, el siguiente paso que debemos aclarar es que se consideran las acciones del bandido multi-brazo. En nuestro contexto de la recomendación de contactos podemos asemejar los acciones a los usuarios a recomendar.

Además, la recompensa obtenida cada vez que se realiza una recomendación (o se acciona un brazo en terminología clásica de bandidos multi-brazo) debe asemejarse a los valoraciones binarios comentados anteriormente. La recompensa por tanto será 1 cuando la recomendación del contacto es aceptada y 0 en caso contrario. Hay que destacar que este es el enfoque habitualmente en la literatura, sin embargo, el experimento de la sección 5.3, consiste en cambiar la noción de recompensa que se obtiene al accionar los diferentes brazos del bandido.

Por último, la definición de recompensa acumulada de un *item*. Se define simplemente como la media de los aciertos entre el total de recomendaciones (aciertos más fallos) de dicho *item*. Este comportamiento se observa en la fórmula 2.5.

$$r_t(i) = \frac{\alpha_t(i)}{\alpha_t(i) + \beta_t(i)} \quad (2.5)$$

donde α_t son las recomendaciones acertadas hasta el tiempo t y β_t son los fallos de recomendar el *item* i hasta tiempo t . Además, para $t = 0$, r_t se puede considerar cualquier constante pero habitualmente se considera 0.

Para resumir esta formalización se muestra la tabla 2.1 de equivalencias entre los conceptos de un sistema de recomendación y los de un bandido multi-brazo.

Una vez detalladas las equivalencias entre sistema de recomendación y Bandido Multi-brazo solo falta detallar el algoritmo que sigue un bandido multi-brazo como se ve a continuación en el algoritmo 2.1:

Sistema de recomendación	Bandidos multi-brazo
<i>item</i>	brazo
valoración	recompensa
valoración promedio	recompensa acumulada

Tabla 2.1: Correspondencia entre sistemas de recomendación y bandidos multi-brazo

```
input :  $U$  conjunto de usuarios
1  foreach  $u \in U$  do
2     $v \leftarrow \text{accionar\_brazo}(u)$ ;
3     $r \leftarrow \text{obtener\_recompensa}(u, v)$ ;
4    actualizar_brazo( $v, r$ );
5  end
```

Algoritmo 2.1: Algoritmo general de recomendación con bandidos para la recomendación de contactos

ALGORITMIA

El desarrollo de algoritmos para la recomendación de contactos ha proliferado enormemente en los últimos años, en los que han aparecido multitud de técnicas y herramientas ya no solo para la recomendación de contactos sino para todo el ámbito de la recomendación.

En esta sección se explica una selección de dichos algoritmos, los cuales serán utilizados posteriormente en los experimentos. Los algoritmos que vamos a tratar a lo largo de este TFM se pueden ver de manera esquemática en la figura 3.1. Concretamente se han elegido 2 grandes tipos de algoritmos. Primero, algoritmos que utilizan la estructura de grafo inherente a la red social. Segundo, un subconjunto de algoritmos de aprendizaje por refuerzo, los Bandidos Multibrazo.

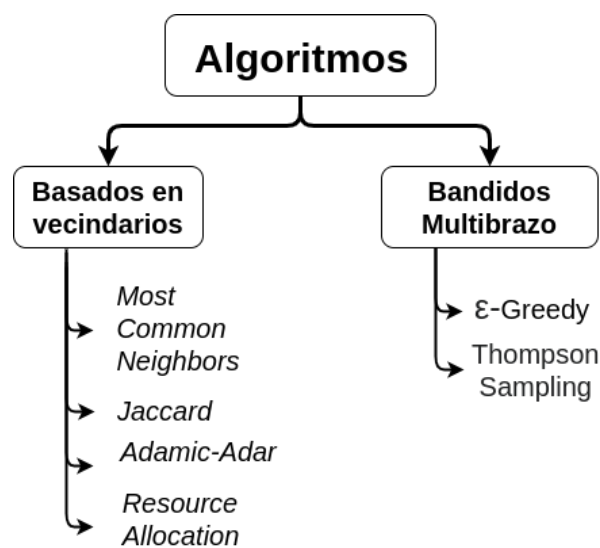


Figura 3.1: Algoritmos de recomendación de contactos tratados en el TFM

El capítulo 3 consta de 2 secciones, en la primera se detallarán los algoritmos que se han implementado para la realización del presente TFM, que se corresponden con los algoritmos basados en vecindarios. En la segunda sección, se expone el funcionamiento de los de aprendizaje por refuerzo, cuya implementación se obtiene de módulos externos.

3.1. Algoritmos implementados

Durante esta sección se detallan los algoritmos implementados en el TFM. Primero, en la sección 3.1.1, se introducen el algoritmo de recomendación aleatoria y la recomendación basada en popularidad, ambos utilizados como líneas base en la literatura. A continuación, en la sección 3.1.2, se introducirán los algoritmos basados en los vecindarios de la red de contactos.

3.1.1. Algoritmos básicos

Tanto el algoritmo de recomendación aleatoria como el basado en popularidad son mayoritariamente utilizados como elementos de control a la hora de evaluar sistemas de recomendación. Destacan por ser sencillos de interpretar e implementar pero dicha simplicidad lleva asociada, sobre todo en la recomendación aleatoria, un perjuicio en los resultados obtenidos.

Aleatorio

La recomendación aleatoria es el algoritmo más utilizado como control puesto que ningún otro sistema más inteligente debe funcionar peor que esta recomendación. En este sentido, es útil puesto que nos muestra la mejora total obtenida por el resto de algoritmos. La recomendación aleatoria se modela mediante una sencilla distribución de probabilidad uniforme.

$$f_u(v) \sim U \quad (3.1)$$

Popularidad

La recomendación por popularidad es la línea base más fácil de justificar. Estamos totalmente acostumbrados a este tipo de recomendación no personalizada. Ejemplos típicos son “Él más vendido” o “Número 1 en cines”. En el ámbito de la recomendación de contactos utilizar este método es algo menos obvio. Concretamente aparece tras la aparición de las redes sociales dirigidas, como puede ser Twitter. Estas redes nos dan la posibilidad de seguir a nuestros personajes famosos favoritos. La fórmula correspondiente a la recomendación por popularidad se muestra en 3.2:

$$f_u = |\Gamma_{in}(v)| \quad (3.2)$$

3.1.2. Algoritmos basados en vecindarios

Para realizar recomendación de contactos en redes sociales se pueden seguir una gran variedad de enfoques. Pero quizás el más directo es utilizar la propia estructura de red que generan los contactos para realizar nuevas recomendaciones. Este es el enfoque que se va a seguir en este TFM. Se utilizarán los contactos de u (usuario al que se va a realizar una recomendación) y los contactos de v (usuario candidato a ser recomendado) para hallar la afinidad que existe entre ellos. En una red no dirigida como Facebook, solo se considera un tipo de contactos de u , llamado en la literatura vecindario de u y denotado $\Gamma(u)$. Sin embargo, las redes dirigidas como Twitter nos permiten realizar distinciones, concretamente:

- $\Gamma_{in}(u)$, conjunto de usuarios que siguen a u .
- $\Gamma_{out}(u)$, conjunto de usuarios seguidos por u .

Además a partir de estos dos vecindarios podemos realizar 1 nueva definición:

- $\Gamma_{und}(u) := \Gamma_{in}(u) \cup \Gamma_{out}(u)$.

Estos vecindarios se muestran en la figura 3.2.

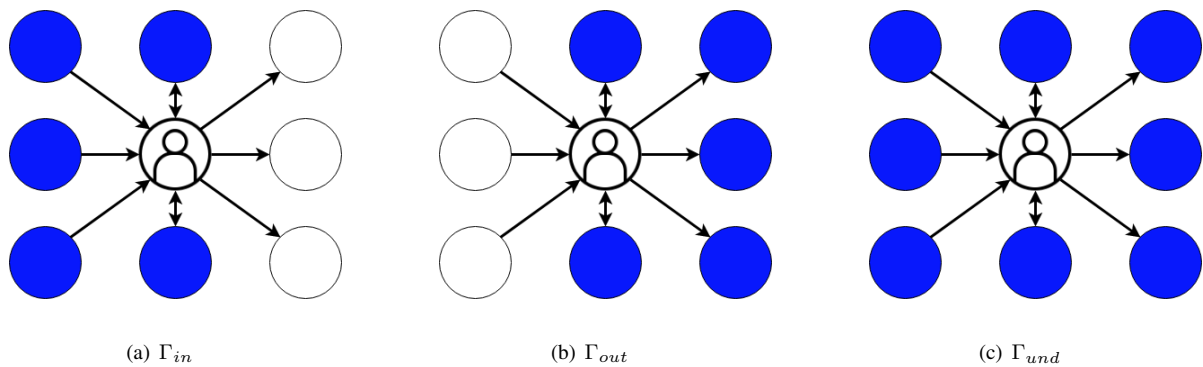


Figura 3.2: Posibles vecindarios para un usuario

Dicho esto, en las siguientes subsecciones se detallan diferentes algoritmos de recomendación basados en vecindarios. En ellos utilizaremos $\Gamma(u)$ para referirnos a un vecindario, pudiendo ser sustituido por cualquiera de los subconjuntos arriba detallados.

Most Common Neighbors

Most Common Neighbors (MCN) [9] es el algoritmo más intuitivo dentro de los algoritmos basados en vecindarios. Su idea es que a mayor número de usuarios en común, más fácil será que los usuarios quieran conectarse entre ellos. Para llevar a cabo esta tarea se sigue la formula 3.3.

$$f_u(v) = |\Gamma(u) \cap \Gamma(v)| \quad (3.3)$$

La principal desventaja de MCN es que favorece de manera injusta a aquellos usuarios que están altamente conectados en el grafo de la red social ya que es más probable que tenga mayor numero de contactos en común con los demás usuarios.

Jaccard

El algoritmo de Jaccard [7] [9] viene motivado por la ventaja que proporciona MCN a los nodos del grafo que estan altamente conectados. Para solucionar dicha desventaja Jaccard pondera los contactos en común que tienen u y v entre la unión de todos los contactos tal y como se muestra en la fórmula 3.4:

$$f_u(v) = \frac{|\Gamma(u) \cap \Gamma(v)|}{|\Gamma(u) \cup \Gamma(v)|} \in [0, 1] \quad (3.4)$$

Adamic-Adar

Adamic-Adar [1] es, de nuevo, una modificación de MCN. MCN cuenta el número de contactos que tienen en común el usuario objetivo u y cada uno de los diferentes usuarios candidatos v . Esto puede entenderse como que cada uno de los usuarios que tienen en común se ponderará de la misma manera.

Este es el matiz que cambia Adamic-Adar. Este algoritmo considera que será más informativo que los usuarios de la intersección tengan pocos contactos. Esto se considera así pues Adamic-Adar entiende que este tipo de usuarios son más selectivos a la hora de entablar amistades por lo que dichas amistades deben tener una mayor consideración. Este calculo se lleva a cabo mediante la fórmula 3.5.

$$f_u(v) = \sum_{w \in (\Gamma(u) \cap \Gamma(v))} \frac{1}{\log|\Gamma(w)|} \quad (3.5)$$

Resource Allocation

Resouce Allocation [11] es una variación de Adamic-Adar. Recibe su nombre de la idea de repartir un recurso entre los diferentes contactos de u y que dichos contactos repitan el proceso. Esto podría repetirse un número cualquiera de veces pero la formula habitual de *Resouce Allocation* mide que cantidad de recurso a llegado a los vecinos de distancia 2 que comparten el usuario objetivo u y cada unos de los usuarios candidatos v como se muestra en la fórmula 3.6.

$$f_u(v) = \frac{1}{|\Gamma(u)|} \sum_{w \in (\Gamma(u) \cap \Gamma(v))} \frac{1}{|\Gamma(w)|} \quad (3.6)$$

La mayor variación con respecto a la fórmula de Adamic-Adar (3.5) es que se ha omitido el logaritmo por lo que tener un gran número de vecinos supone una mayor penalización.

3.2. Algoritmos de implementación externa: Bandidos Multi-brazo

Además de los algoritmos tratados en la sección anterior, durante este TFM se ha hecho uso de algoritmos de Bandidos Multi-brazo. Concretamente se han utilizado los algoritmos de ϵ -greedy y *Thompson Sampling* ambos implementados por Esther López Ramos y disponibles en su [github](#).

3.2.1. ϵ -Greedy

El algoritmo de ϵ -Greedy es uno de los más sencillos dentro de los Bandidos Multi-brazo. En primer se debe fijar el parámetro $\epsilon \in [0, 1]$, este sirve para diferenciar las etapas de exploración y explotación. Una vez fijado, con probabilidad ϵ se realizará la etapa de exploración y con probabilidad $1-\epsilon$ se explotará el *item* con mayor recompensa hasta el momento. Este proceso se formula en 3.7.

$$i_t = \begin{cases} i \in I \text{ uniformemente al azar,} & \epsilon \\ \operatorname{argmax}_{i \in I} [r_t(i)], & 1 - \epsilon \end{cases} \quad (3.7)$$

donde r se corresponde con la función recompensa definida en 2.5.

3.2.2. *Thompson Sampling*

Como se ha comentado anteriormente la principal diferencia de los distintos bandidos es su gestión de la exploración-explotación. Mientras que el algoritmo de ϵ -Greedy realiza esta tarea simplemente diferenciando la exploración de explotación mediante un ϵ aleatorio, *Thompson Sampling* toma un enfoque algo más complejo. Concretamente se realiza un muestreo para obtener la recompensa media de cada brazo. La función recompensa r (ver 2.5) se ha definido como los aciertos obtenidos (α) entre los intentos realizados, es decir, los aciertos más los fallos ($\alpha + \beta$ respectivamente). Donde α y β solo pueden ser valores binarios. Esto nos lleva a que la media de la función recompensa sigue una distribución de Bernoulli de un cierto parámetro p (que además coincide con la media de la Bernoulli). Por tanto, el parámetro p puede obtenerse a través de una distribución Beta (por ser la conjugada de la Bernoulli) de parámetros α y β . A mayor número de intentos ($\alpha + \beta$) mayor será la certeza de que la media p es $\frac{\alpha}{\alpha + \beta}$. Por tanto, *Thompson Sampling* utiliza la fórmula 3.8 para seleccionar un *item* en cada iteración.

$$i_t = \operatorname{argmax}_{i \in I} [x \leftarrow B(\alpha_t(i) + \alpha_0, \beta_t(i) + \alpha_0)] \quad (3.8)$$

donde B denota una distribución Beta cuyos parámetros son unos aciertos y fallos iniciales (α_0 y β_0) comunes a todos los *items* y después se suman los aciertos y fallos propios de cada *item* i hasta el momento t . Y donde x es un número aleatorio obtenido de dicha distribución Beta.

EVALUACIÓN Y MÉTRICAS

A día de hoy no existe un método cerrado para evaluar un sistema de recomendación. La evaluación de los mismos puede hacerse tanto *online* como *offline*. La primera se realiza en los sistemas en producción y resulta de interés puesto que permite obtener la opinión de los usuarios de la plataforma. Sin embargo, los requisitos son difíciles de alcanzar, ya que debemos tener acceso a dicho sistema en producción. En nuestro caso concreto, realizaremos experimentos con datos de la red social Twitter, por lo que evaluar *online* no es una opción (ya que no tenemos acceso a las estructuras internas de Twitter) y por tanto, en este TFM realizaremos una evaluación *offline*.

La evaluación *offline* requiere tener un conjunto de datos descargados de antemano al que se le aplica un tipo de partición *train - test*. Estas particiones pueden ser de distinto tipo, normalmente se diferencian entre particiones aleatorias, por usuarios, por *items* o temporales, entre otras. En nuestro caso la partición, debido a que utilizamos datos reales de la red de Twitter, la partición será temporal. Esto se debe a que queremos un estado de la red realista a la hora de realizar recomendaciones y queremos comprobar si a *posteriori* ese enlace se dio realmente en la red. Hay que destacar que si bien el corte temporal de los datos es el necesario para este tipo de experimentos, utilizarlo conlleva alguna desventaja ya que no es posible realizar una validación cruzada que de cierta suavidad a los datos sino que se vuelve necesario realizar una validación simple por lo que podría llegar a producirse un sobreajuste en los modelos de recomendación de contactos.

El TFM quiere investigar como evoluciona la red social desde un punto de vista global al utilizar un recomendador de contactos. Por ello ha continuación se detallan las diferentes métricas de grafo que se van a utilizar para medir diferentes aspectos de la red social y la evolución que estos experimentan según avanza el tiempo. Concretamente este capítulo consta de 4 secciones en las que se verán la métricas basadas en distancias (4.1), las métricas de diversidad estructural (4.2), las métricas de novedad (4.3) y por ultimo las métricas de diversidad (4.4).

4.1. Métricas basadas en distancia

Con la evolución de la red, ya sea por la intervención de un sistema de recomendación de contactos o por otros métodos como la búsqueda manual por parte de los usuarios de conocidos a los que añaden dentro de la red social, las distancias entre todos los usuarios de la red tiendan a ir reduciéndose. Es por esto que medir las distancias entre los nodos resulta de interés para la evolución del grafo en su conjunto.

Cuando hablamos de distancia entre 2 puntos, se entiende, mientras que no se especifique lo contrario, que se trata de la distancia mínima entre dichos puntos. Para visualizar esto, observemos la figura 4.1. En ella vemos como existen 2 caminos posibles para llegar de U a V, en el camino rojo debemos pasar por 2 nodos antes de llegar a V, o lo que es lo mismo, debemos pasar por 3 aristas del grafo. Esto implica que por el camino rojo tenemos distancia 3. Sin embargo, el camino verde solo consta de 2 aristas. Es por esto que el camino verde se corresponde con la distancia (mínima) entre U y V. Además, si no existe forma de llegar a U a V se considera que distancia es ∞ .

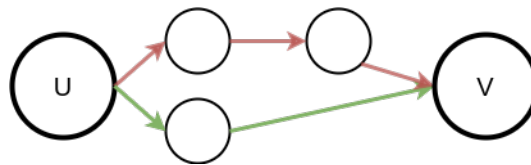


Figura 4.1: Ejemplo distancia $\delta(u, v)$

Concretamente en este TFM hemos medido las siguientes métricas de distancias: Reciprocal diameter (RD) y Reciprocal average eccentricity (RAE).

4.1.1. Reciprocal Diameter

En teoría de grafos se define el diámetro de un grafo G como el camino mínimo más largo dentro de la red, tal y como se muestra en la fórmula 4.1.

$$\text{Dimetro}(G) = \max\{\delta(u, v)\} \forall u, v \in \mathcal{U} \quad (4.1)$$

Un ejemplo del diámetro lo vemos en la figura 4.2. El grafo de la figura tiene un diámetro de 4, obtenido al calcular la distancia del camino de color verde. Además la figura nos muestra como puede haber diferentes pares de nodos que nos den el mismo diámetro. Ya que el diámetro en este caso puede ser atribuido al par de nodos (A,Z) o bien al par (B,Z).

Por la definición de distancia y su implicación directa sobre la definición de diámetro, podríamos obtener que el diámetro de un grafo es ∞ si el grafo es desconexo. Esto resulta incomodo puesto que

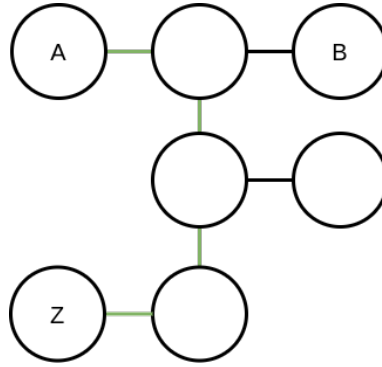


Figura 4.2: Ejemplo diámetro de un grafo G

en la mayoría de los casos se manejan grafos desconexos y por tanto el diámetro sería ∞ un gran número de ocasiones. Es por esto que para métricas como *Reciprocal Diameter* (RD) consideramos que el diámetro se corresponde con el diámetro de la mayor componente conexa del grafo. Por tanto, definimos *Reciprocal diameter* como se muestra en la fórmula 4.2.

$$RD(G) = \frac{1}{\text{Diametro}(G)} \quad (4.2)$$

RD toma valores entre $(0,1]$, siendo 1 exclusivamente si el diámetro del grafo es 1, o lo que es lo mismo, todos los nodos (dentro de cada componente conexa) están conectados entre si. Cabe destacar que RD es una métrica que sufre pocas variaciones ya que resulta complicado aumentar el diámetro de un grafo mediante recomendación si el escenario es realista. Esta es la principal razón de definir otras métricas como *Reciprocal Average Eccentricity*, que se define a continuación en la sección 4.1.2.

4.1.2. *Reciprocal Average Eccentricity*

La excentricidad de un nodo u se define como el nodo más lejano que se puede alcanzar. Esto se muestra en la fórmula 4.3.

$$ecc(u) = \max\{\delta(u, v) \mid \forall v \in \mathcal{U} \mid \delta(u, v) < \infty\} \quad (4.3)$$

Por tanto, otra métrica que puede resultar interesante es el promedio de las excentricidades de los nodos del grafo, por ello se define *Reciprocal Average Eccentricity* (RAE) según la fórmula 4.4.

$$RAE(u) = \frac{|\mathcal{U}|}{\sum_{u \in \mathcal{U}} ecc(u)} \quad (4.4)$$

Al igual que RD, esta métrica toma valores pertenecientes al intervalo (0,1]. Obteniendo el valor 1 cuando todos los nodos se encuentran conectados unos a otros directamente y valores cercanos a 0 si existen nodos muy lejanos dentro del grafo. Así, esta métrica se puede considerar una versión menos rígida que RD.

4.2. Métricas de diversidad estructural

A lo largo de esta sección se van a detallar las diferentes métricas implementadas para medir la diversidad estructural de la red social (y su correspondiente grafo asociado) para medir el efecto de las recomendaciones. Gran parte de estas métricas son habituales en teoría de grafos, pero merece la pena mencionarlas, analizarlas y comentar sus matices dentro del área de recomendación de contactos.

4.2.1. Degree Gini Complement

Esta es la métrica de diversidad estructural más sencilla. Su objetivo es resumir la distribución del grado de los nodos del grafo en un solo número.

$$DGC(G) = 1 - \frac{1}{|\mathcal{U}| - 1} \sum_{i=1}^{|\mathcal{U}|} \frac{|\Gamma(u_i)|}{|E|} \quad (4.5)$$

En la fórmula 4.5 se deben considerar los usuarios u_i en orden creciente de vecinos. Esto es necesario para captar la distribución de los datos. La fórmula devuelve valores en el intervalo [0,1]. Los valores cercanos a 0 indican que existen nodos que acaparan la mayoría de las conexiones del grafo mientras que si el resultado de aplicar DGC es cercano a 1 se observa como los enlaces de la red social están cada vez mas equidistribuidos. La fórmula 4.5 se corresponde con un grafo no dirigido. Sin embargo, esta formula es fácilmente aplicable a grafos dirigidos. Simplemente debemos considerar si queremos utilizar el vecindario de entrada o de salida (o la unión de ambos, en cuyo caso seria equivalente a la formula para grafos no dirigidos,)

DGC es la métrica mas sencilla para la diversidad estructural. Las demás métricas se pueden meter en 2 grandes grupos. Las métricas de carácter Local 4.2.2 y las métricas de diversidad global 4.2.3.

4.2.2. Métricas locales en diversidad

Para poder entender las métricas de carácter local primero es necesario discutir lo que se define como un enlace débil. Se utiliza el calificativo de débil puesto que se entiende que si ese enlace no existiese

la red perdería ciertas conexiones. Así, se ponen en consideración cuestiones como la redundancia que existe dentro de la red. Un ejemplo de enlace débil lo vemos en la figura 4.3 donde se muestra en verde ejemplos de enlaces débiles. Se consideran débiles dichos enlaces puesto que no sería posible llegar de A a Z si cualquiera de esos enlaces fuese eliminado del grafo.

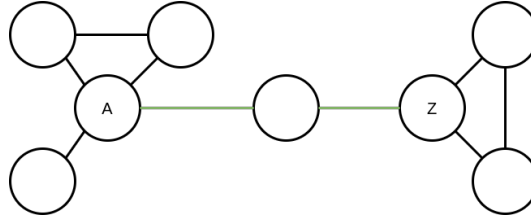


Figura 4.3: Ejemplo enlace débil

Average Edge Weakness

Sin embargo, el ejemplo 4.3 puede resultar demasiado restrictivo puesto que se considera enlace débil únicamente si existe un único camino entre 2 nodos. Es por esto que se han definido otras nociones para hacer más flexible esta noción de redundancia, entre ellas destaca *Embeddedness*, con la que se consigue pasar de la consideración binaria al intervalo $[0,1]$.

$$Embeddedness(u, v) = \frac{|\Gamma_{out}(u) \cap \Gamma_{in}(v)|}{|\Gamma_{out}(u) \cup \Gamma_{in}(v)|} \quad (4.6)$$

Utilizando *Embeddedness* como base podemos definir la capacidad que tiene un sistema de recomendación de recomendar enlaces débiles mediante la métrica *Average Edge Weakness*:

$$AEW(u, v) = \frac{1}{|\hat{E}|} \sum_{(u,v) \in \hat{E}} (1 - Embeddedness(u, v)) \quad (4.7)$$

donde \hat{E} se corresponde con los nuevos enlaces que el sistema de recomendación predijo correctamente. Esta métrica obtiene valores en $[0,1]$. Si tenemos valores cercanos a 1 significa que se recomiendan con éxito un mayor número de enlaces débiles, lo que se traduce en una mayor diversidad en la red gracias a la recomendación.

Clustering Coeficient Complement

El coeficiente de clustering es una de las métricas clásicas para medir la redundancia de enlaces en un grafo. Consta de varias definiciones posibles y se ha optado por la que se muestra en la fórmula 4.8. Esta formulación considera el número de triángulos cerrados que se generan en comparación con el número de tripletas (abiertas o no).

$$CCC(G) = 1 - \frac{|\{(u, v, w)\}|(u, v)(v, w), (w, u) \in E|}{|\{(u, v, w)\}|(u, v)(v, w) \in E|} \quad (4.8)$$

4.2.3. Métricas globales en diversidad

Del mismo modo que se define un enlace débil a nivel local, también se considera su definición a nivel global para un grafo. De nuevo se define como una conexión imprescindible para conectar 2 partes de la red, sin embargo esta vez, en lugar de conectar usuarios conecta componentes conexas del grafo. Por tanto, el enlace débil en el ámbito global vuelve a considerar nociones de conexión y redundancia del grafo.

Community Edge Gini Complement

Con esta métrica pretendemos medir como se distribuyen los enlaces débiles entre diferentes componentes conexas. Así, no nos limitamos a considerar el número de nuevos enlaces que conectan componentes sino que consideramos si estos enlaces se producen entre unas comunidades específicas o están equidistribuidos entre los diferentes pares de componentes. La fórmula para *Community Edge Gini Complement* se muestra en 4.9.

$$CEGC(G|\mathcal{C}) = 1 - \frac{1}{N-1} \sum_{i=1}^N (2i - N - 1) \left(\frac{x_i}{|E|} \right) \quad (4.9)$$

Donde \mathcal{C} se corresponde con un conjunto de componentes conexas, N es el número de conexiones posibles entre dichas componentes sin repetición + 1, por considerar también los enlaces internos en cada componente, y donde x_i es el número de enlaces que van de una componente a otra y también la suma de todos los enlaces que se quedan en las componentes conexas.

Del mismo modo que ocurría en otras métricas donde se utiliza el índice Gini, en esta los x_i deben ir ordenado en orden creciente, para así captar la distribución que estos generan. Los valores de la métrica van del 0 al 1. Se obtiene el valor 0 si todos los enlaces débiles son de una comunidad a otra o si no existe ningún enlace débil. Por el contrario se alcanza el valor 1 cuando existe el mismo número de enlaces débiles entre cada par de comunidades.

4.3. Métricas de novedad

La novedad ha sido desde el inicio de la recomendación uno de las principales cuestiones tratadas. Esto se debe a que recomendar lleva intrínseco un carácter de búsqueda de novedades. Un ejemplo

de esto se visualiza fácilmente con la recomendación de películas. Se entiende rápidamente que si tenemos un sistema de recomendación que principalmente recomienda películas de conocido prestigio, como podrían ser “Titanic”, “Avatar” o “Vengadores:End game” entre muchos otros títulos, pues puede que tenga un gran acierto puesto que son grandes títulos que merece la pena ver, pero lo más probable es que esas películas ya las haya visto el usuario al que se las estas recomendando. Es por esto que estas recomendaciones serán ignoradas por el usuario. Y esto tiene un efecto negativo en el sistema puesto que recomendar esas grandes películas supone que quedan ocultas otras que es probable que también fuesen de interés para el usuario.

Al llevarnos esta idea al ámbito de la recomendación de contactos, la similitud que tenemos es que debemos intentar recomendar usuarios con pocos contactos. Justo en este sentido surge la métrica *Long Tail Novelty*.

4.3.1. Long Tail Novelty

Long Tail Novelty pretende reflejar la probabilidad media de que la persona que recomendamos resulte novedosa para el usuario. Esta idea puede formalizarse como que el usuario u no se encuentre en el vecindario de entrada de v . Por tanto podemos definir LTN como se muestra en 4.10.

$$LTN(\hat{E}|G) = \frac{1}{|\hat{E}|} \sum_{(u,v) \in \hat{E}} \left(1 - \frac{|\Gamma_{in}(v)|}{|\mathcal{U}|\right) \quad (4.10)$$

Donde \hat{E} simboliza los enlaces recomendados correctamente por el sistema. Esta métrica tiene un carácter global puesto que se considera la novedad que aporta el usuario recomendado v independientemente del usuario al que se esta recomendando. Existen métricas que personalizan a LTN incorporando al usuario u . Sin embargo puesto que en este trabajo se pretende ver la red en su conjunto no es necesario recurrir a fórmulas más complejas.

4.4. Métricas de diversidad

La diversidad en la recomendación es uno de los temas que más se esta tratando a día de hoy en el estado del arte. Esto se debe a que cuando la recomendación se iniciaba como campo se pretendía obtener maximizar el acierto de las recomendaciones realizadas y esto llevaba a métodos de que, en gran medida, encasillaban al usuario que recibía la recomendación. En la última década los esfuerzos dentro de la comunidad científica han crecido para solucionar este problema.

4.4.1. *Prediction Gini Complement*

Para medir la diversidad se va a volver a utilizar el índice de Gini. Concretamente se va a observar la distribución de las recomendaciones. Tendremos un índice de Gini de 1 si todos los usuarios son recomendados el mismo numero de veces y por el contrario, obtendremos valores cercanos 0 si existen ciertos usuarios que están siendo recomendados mayoritariamente. Se ve el diseño de esta métrica en la fórmula 4.11.

$$PGC(G) = 1 - \frac{1}{|\mathcal{U}| - 1} \sum_{i=1}^{|\mathcal{U}|} (2i - |\mathcal{U}| - 1) \frac{|\hat{\Gamma}_{in}(v_i)|}{|\hat{E}|} \quad (4.11)$$

Como en el resto de índices Gini, los elementos v_i deben ir ordenados, en este caso por orden creciente de $|\hat{\Gamma}_{in}(v_i)|$.

EXPERIMENTOS

El capítulo de experimentos consta de 3 secciones. En primer lugar se detallan los datos utilizados en la sección 5.1. Posteriormente se realiza la comparativa de algoritmos en la sección 5.2. En esta sección veremos como evolucionan las características de la red al aplicar diferentes algoritmos de recomendación, ya sean algoritmos basados en vecindarios o algoritmos de bandidos multi-brazo. Por último, la sección 5.3 muestra como varían las características de la red al modificar los bandidos utilizando novedosas funciones de acierto.

5.1. Configuración del experimento

5.1.1. Datos utilizados

Los experimentos realizados durante el TFM utilizan datos reales de la red social (dirigida) Twitter. Es un conjunto de datos de los muchos tratados en [10, chap. 7.1]. Estos datos se obtuvieron mediante la API pública, haciendo uso de un procedimiento de bola de nieve [5]. Concretamente el proceso utilizado es el siguiente:

En primer lugar se seleccionó un usuario semilla u . Se recogieron todas las menciones y retweets que u hacía de otros v . Este procedimiento se repitió con todos los usuarios v hasta alcanzar un número de 10.000 usuarios. Una vez alcanzado dicho número se obtuvieron los vecinos de todos los usuarios y se guardaron las relaciones de amistad existentes entre dichos 10.000 usuarios.

Para los experimentos estos datos se han dividido en dos conjuntos, conjunto de entrenamiento y conjunto de test. La división de estos datos se realiza siguiendo su orden natural de creación en la red social, dado que realizar un tipo de corte que no fuese temporal resultaría en la creación de situaciones ficticias que podrían simular escenarios inverosímiles. Concretamente, el conjunto de test se obtiene repitiendo este proceso 1 mes después obtener el conjunto de train y viendo como se han incrementado las conexiones entre los 10.000 usuarios seleccionados. La separación temporal realizada se detalla en la tabla 5.1.

	No. usuarios	No. aristas
Train	9.964	475,530
Test	8,180	98,519
Total	10.000	582.172

Tabla 5.1: Detalles de la red generada por los datos

5.2. Comparativa de algoritmos

Este TFM se centra en el análisis de la evolución de las redes sociales cuando estas utilizar un sistema de recomendación de contactos. Por ello, el primer experimento realizado consiste en medir las métricas expuestas en el capítulo 4 para las mejores configuraciones de los algoritmos del capítulo 3 del presente trabajo. Estas configuraciones óptimas se han obtenido de estudios previos (concretamente [6] para los algoritmos basados en vecindarios y [8] en los relacionados con bandidos multi-brazo). Dichos estudios no maximizaban ninguna de las métricas que se tratan en el capítulo 4 sino que su objetivo era maximizar la métrica de Precisión. La precisión mide el número de aciertos al recomendar y lo divide por el número de recomendaciones realizadas.

Se han utilizado las configuraciones óptimas para Precisión (mostradas en la tabla 5.2) para recomendar puesto que en un sistema de recomendación en producción esta es la característica principal a obtener, acierto. Por consiguiente, resulta interesante medir el impacto que tienen los sistemas de recomendación con mayor Precisión sobre la evolución de la red social.

Algoritmo	Parametros
MCN	$\Gamma_{und}(u) \Gamma_{in}(v)$
Jaccard	$\Gamma_{und}(u) \Gamma_{in}(v)$
Adamic-Adar	$\Gamma_{und}(u) \Gamma_{in}(v) \Gamma_{und}(w)$
Resource Allocation	$\Gamma_{und}(u) \Gamma_{in}(v) \Gamma_{und}(w)$
ϵ -Greedy	$\epsilon = 0.2$
Thompson Sampling	No tiene parámetros

Tabla 5.2: Mejores configuraciones para los algoritmos basados en vecindarios para la recomendación estática

Con las configuraciones ya seleccionadas para cada algoritmo, ahora toca realizar una comparativa para dichas configuraciones en la que se aplique la perspectiva cíclica, llegando a realizar 1000 iteraciones. Concretamente, durante el experimento se recomienda a todos los usuarios 1000 veces, pero solo se evalúan las métricas de grafo cada 10 iteraciones. Esto se hace puesto que estas métricas son computacionalmente costosas y observar los cambios cada iteración puede resultar poco significativo.

A continuación se muestran los resultados por cada métrica expuesta en el capítulo 4 salvo Reciprocal Diameter, puesto que esta métrica no sufre cambios para ningún algoritmo en las 1000 iteraciones

realizadas obteniendo siempre un valor de 0.1.

Reciprocal Average Eccentricity

La figura 5.1 muestra la evolución de la métrica de *Reciprocal Average Eccentricity* en función del algoritmo de recomendación utilizado. Se observan claramente 2 grupos, en primer lugar los algoritmos basados en vecindarios y con un peor resultado los bandidos multibrazo.

Recordemos que esta métrica alcanza el valor 1 si todos los nodos están conectados entre si y es cercana a 0 cuando se deben recorrer grandes distancias para llegar de un nodo a otro. A la vista de los resultados, Twitter no es una red en la que los nodos estén muy interconectados directamente entre si.

Además de esta gráfica destaca que todos los algoritmos menos *Thompson Sampling* alcanzan un mínimo local en las primeras iteraciones y posteriormente se estabilizan en torno a un valor. Esto indica que la recomendación de contactos a partir de dicho mínimo conecta varias componentes disconexas y por tanto aumenta el RAE.

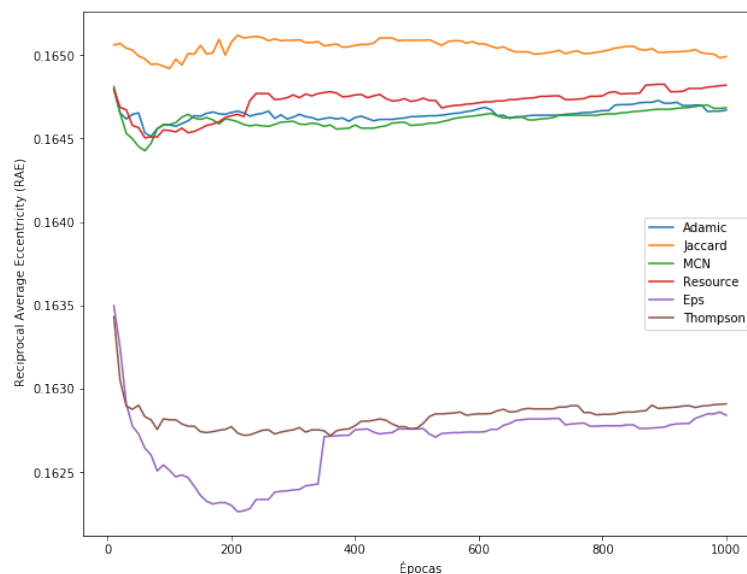


Figura 5.1: Comparativa de *Reciprocal Average Eccentricity* para los diferentes algoritmos de recomendación

Degree Gini Complement

La figura 5.2 se corresponde con los resultados para *Degree Gini Complement*. Se observa claramente un descenso paulatino de la métrica conforme avanzan las iteraciones de recomendación. Este comportamiento es habitual en las redes sociales, especialmente en las dirigidas. Recordemos que DGC obtiene valores cercanos a 1 si las conexiones entre los nodos del grafo se están equidistribuidas y valores próximos a 0 si existen nodos que acaparan gran parte de las conexiones. Es habitual que los

algoritmos recomiendan a usuarios populares y por esto las curvas obtenidas midiendo DGC tienen pendiente negativa. Cabe resaltar dos cuestiones sobre esto. En primer lugar, la curva de Jaccard tiene mucha menor pendiente que las demás, esto se debe a que Jaccard es un algoritmo que, por construcción, penaliza a los usuarios populares y por tanto obtiene un DGC mayor. Segundo, los bandidos multibrazo tiene pendientes mayores a los demás algoritmos. Esto también es un comportamiento natural ya que los bandidos multibrazo tienden a recomendar un mayor número de veces aquellos usuarios que han dado buenos resultados en recomendaciones anteriores y esto, principalmente, son usuarios altamente conectados.

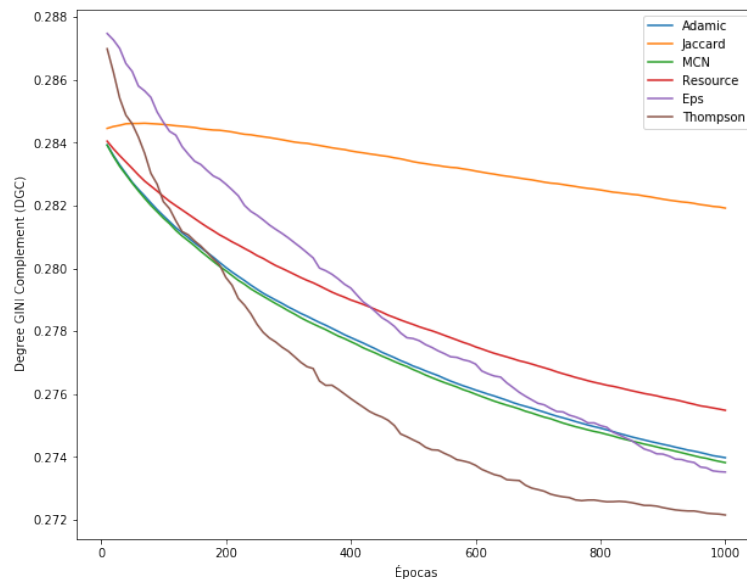


Figura 5.2: Comparativa de *Degree Gini Complement* para los diferentes algoritmos de recomendación

Average Edge Weakness

AEW mide la diversidad de la red mediante las conexiones de enlaces débiles que se han recomendado con éxito. Valores cercanos a 1 implican una mayor diversidad en la recomendación ya que se han recomendado facetas de los usuarios que antes no estaban plasmadas en el grafo de la red social.

La figura 5.3 muestra los resultados obtenidos. La principal característica de este grafo es que todos los algoritmos tienden al mismo valor pero la métrica inicialmente es distinta. Destaca sobre todo Jaccard que obtiene valores muy bajos en un primer momento y rápidamente alcanza al resto de algoritmos basados en vecindarios. Por otro lado tenemos a los bandidos multibrazo. En esta gráfica se observa con claridad su comportamiento de exploración-explotación, en un primer momento los bandidos multibrazo obtienen mayor diversidad puesto que están buscando que usuarios son los mejores (los que son más aceptados) a la hora de recomendar. Posteriormente, cuando ya tienen evidencia suficiente para recomendar siempre a ciertos usuarios los bandidos multibrazo pierden AEW.

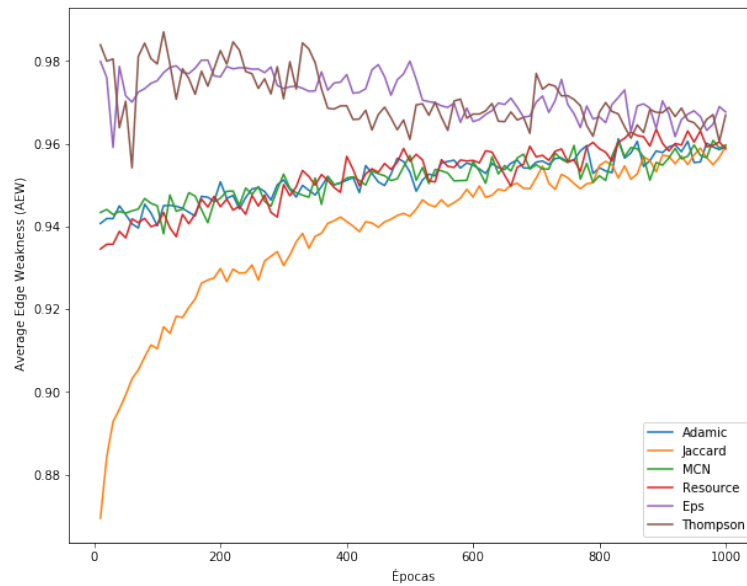


Figura 5.3: Comparativa de *Average Edge Weakness* para los diferentes algoritmos de recomendación

Clustering Coeficient Complement

El coeficiente de clustering mide la interconexión de los nodos del grafo. Es típicamente baja en las redes sociales y conforme pasa el tiempo tiende a degradarse (acercarse a 0). La figura 5.4 muestra los resultados.

Destaca principalmente el cambio de tendencia que obtienen los bandidos multibrazo, que incluso llegan a aumentar la métrica en las primeras épocas, aunque luego se observa de nuevo una pendiente negativa. Este comportamiento resulta significativo puesto que implica que durante la etapa principal de exploración los bandidos consiguen un mayor número de tripletas cerradas.

Comunity Edge Gini Complement

CEGC mide la distribución de las nuevas conexiones que ocurren entre nodos de diferentes comunidades. Recordemos que se obtendría valor 1 cuando se producen el mismo numero de conexiones dentro de cada comunidad como los que se producen entre las diferentes comunidades. Por el contrario, los valores cercanos a 0 ocurre cuando, o no hay conexiones entre comunidades, o todas las conexiones entre diferentes comunidades se realizan entre 2 comunidades concretas.

La figura 5.5 muestra los resultados para cada uno de los algoritmos estudiados de CEGC. Esta gráfica destaca por tener grandes saltos para todos los algoritmos, estos saltos se producen cuando hay un cambio en el número de comunidades de los que consta la red. Concretamente ocurre con la unión de 2 comunidades, pasando de haber 6 comunidades a tener 5.

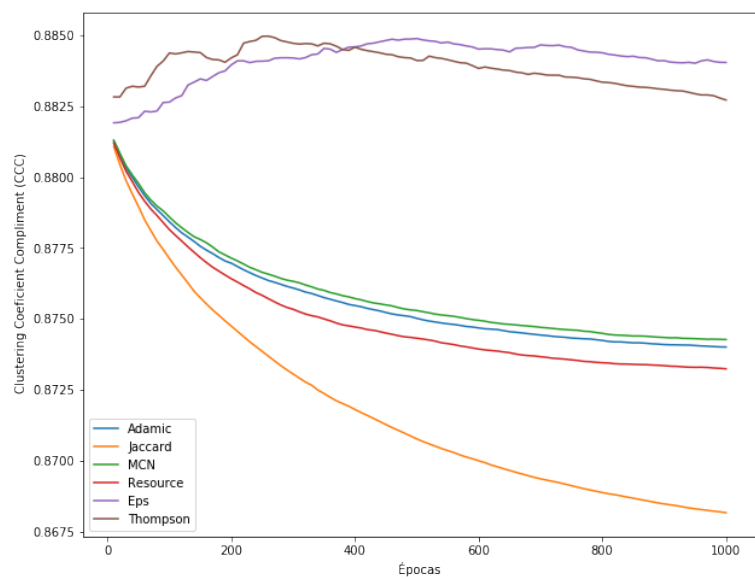


Figura 5.4: Comparativa de *Clustering Coefficient Complement* para los diferentes algoritmos de recomendación

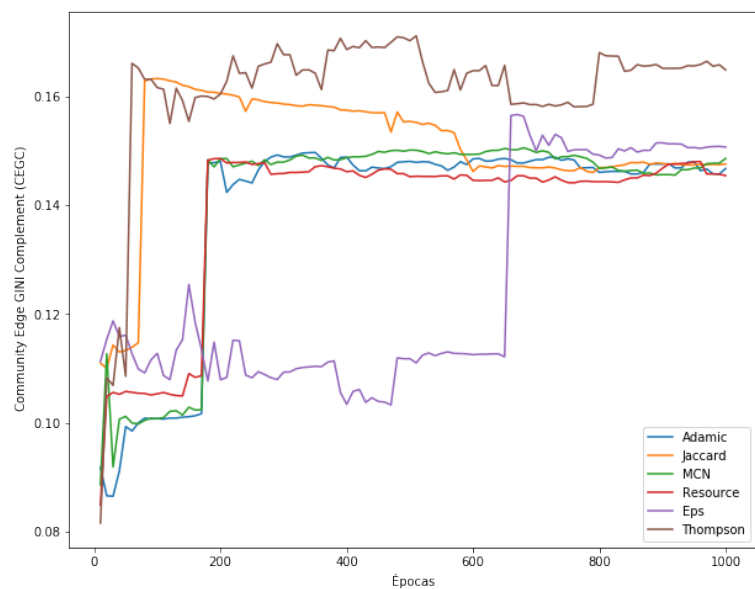


Figura 5.5: Comparativa de *Community Edge Gini Complement* para los diferentes algoritmos de recomendación

Long Tail Novelty

LTN mide la novedad de las recomendaciones, las oscilaciones que se observan en la gráfica 5.6 son normales ya que esta métrica observa la novedad que aporta cada iteración independientemente de las demás, por lo que las oscilaciones no son de extrañar.

Se observa que todos los algoritmos menos Thompson Sampling tienden a un valor. Además, Jaccard y Epsilon greedy se comportan de una manera similar, mientras que Thompson Sampling empieza como los demás algoritmos basados en vecindarios y luego obtiene una ventaja considerable con respecto al resto de algoritmos.

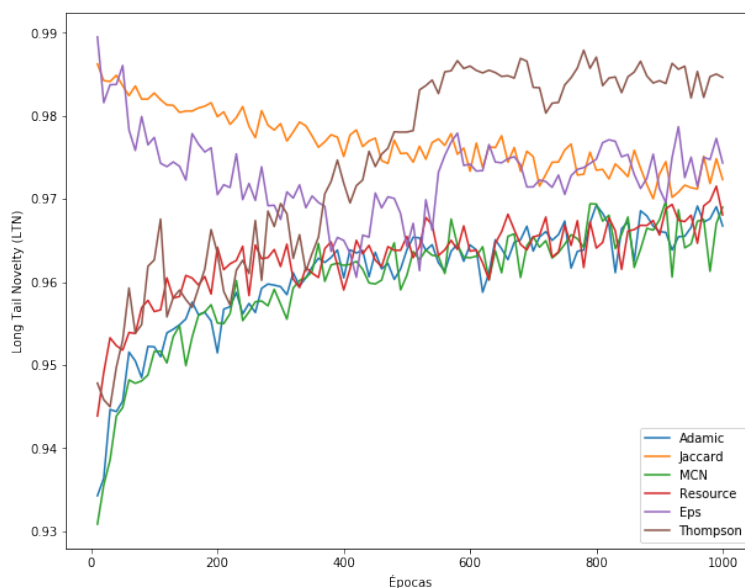


Figura 5.6: Comparativa de *Long Tail Novelty* para los diferentes algoritmos de recomendación

Prediction Gini Complement

PGC mide la distribución de los *items* a recomendar. Los valores cercanos a 1 se obtienen si todos los *items* son recomendados en igual medida y por el contrario, si se tiende a recomendar unos *items* de manera especial obtenemos valores cercanos a 0. Con esta consideración en mente la figura 5.7 debería de resultarnos natural. Mientras que los algoritmos basados en vecindarios recomiendan de manera más equitativa a los diferentes usuarios de la red social, se observa claramente como los bandidos multibrazo explotan aquellos *items* que consideran los mejores.

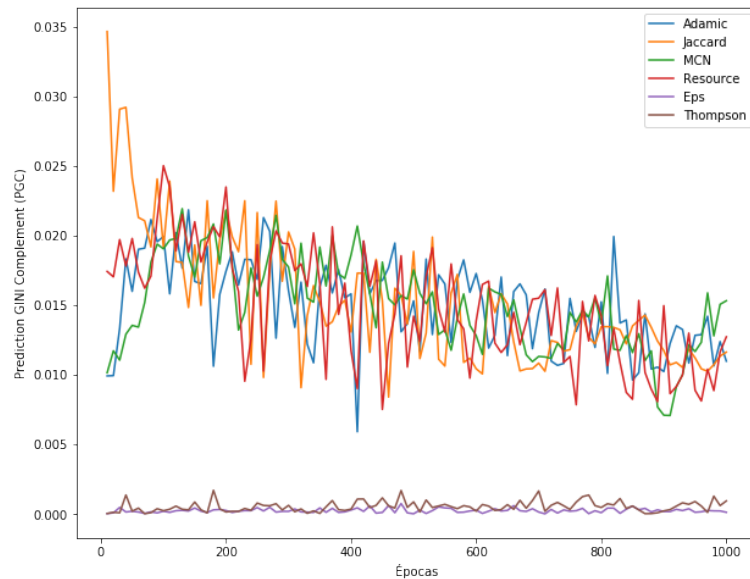


Figura 5.7: Comparativa de *Prediction Gini Complement* para los diferentes algoritmos de recomendación

5.3. Nuevas funciones de recompensa en Bandidos Multi-brazo

Durante esta sección se va a tratar el experimento más relevante del TFM. Concretamente se va a probar la perspectiva novedosa de variar las funciones de recompensa explicadas en la sección 2.4.2. Para realizar este cambio principalmente es necesario modificar la noción de acierto. En el modelo estándar, para considerar una recomendación como acierto basta con que el usuario aceptase la recomendación. El cambio realizado en este experimento tiene una naturaleza más restrictiva. Ahora no solo bastará con que la recomendación sea aceptada, la métrica de grafo resultante de aceptar dicha recomendación debe ser mayor que la que se obtenía anteriormente.

A continuación se muestran los resultados de considerar varias de las métricas de grafo expuestas en el capítulo 4 como funciones de recompensa para los bandidos multi-brazo.

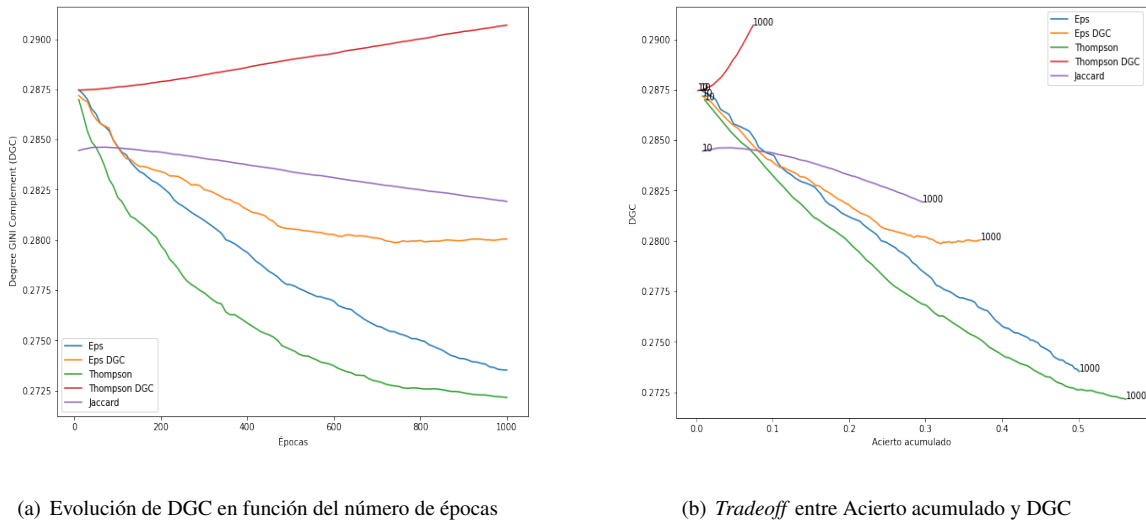
Degree Gini Complement como acierto

En primer lugar consideraremos la métrica de DGC para incorporarla a la noción de acierto de los bandidos multibrazo. Conseguir aumentar DGC conlleva un reparto más equitativo de las conexiones.

La figura 5.8(a) muestra la evolución de la métrica DGC cuando se realizan versiones optimizadas de *Epsilon Greedy* y *Thompson Sampling*, y compara estos resultados con los ya obtenidos para sus versiones estándar y el mejor de los algoritmos basados en vecindarios.

Sin embargo esta comparativa puede resultar sesgada puesto que no sabemos cual es la pérdida

de acierto al recomendar. Por ello la figura 5.8(b) muestra una gráfica de *tradeoff* entre el acierto acumulado por el recomendador y DGC obtenido.



(a) Evolución de DGC en función del número de épocas

(b) *Tradeoff* entre Acierto acumulado y DGC

Figura 5.8: Resultados de utilizar DGC como acierto en los bandidos multi-brazo

Al mirar en su conjunto las figuras 5.8 se observa como un aumento de DGC se corresponde con un pérdida en el acierto, llegando al caso extremo de *Thompson Sampling* optimizado para DGC.

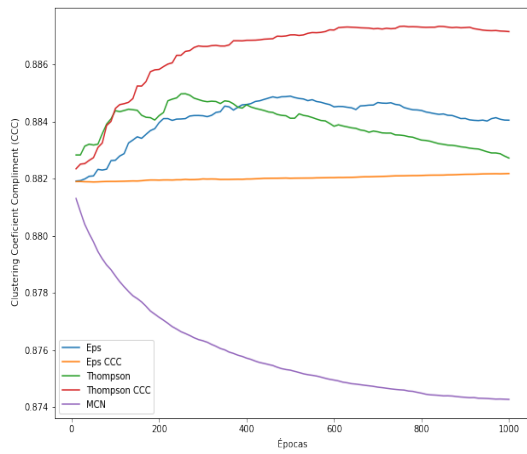
Clustering Coeficient Complement como acierto

Repitamos el análisis anterior ahora para la métrica de CCC. En la figura 5.9(a) se observa como la optimización de *Thompson* mejora considerablemente las medidas obtenidas para el coeficiente de clustering. Por el contrario, *Epsilon Greedy* empeora los resultados al pasar a la optimización, sin embargo no se cambia a la tendencia de los algoritmos basados en vecindarios.

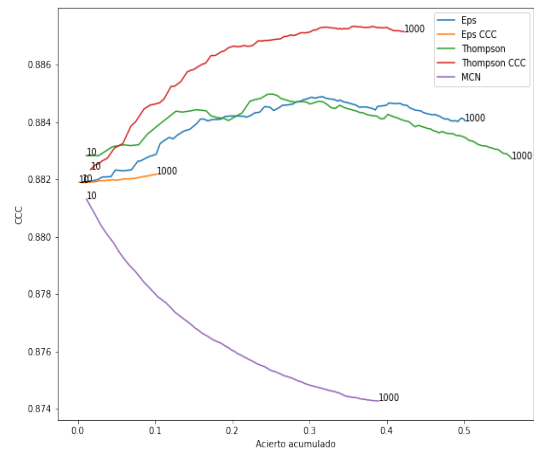
Al fijarnos en la figura 5.9(b) se observa además como *Thompson Sampling* no ha sufrido una penalización exagerada en el acierto pero *Epsilon* si. Esto puede deberse simplemente a la aleatoriedad que tiene este algoritmo a la hora de realizar la exploración de los *items*. Por ello sería conveniente confirmar esta medición en trabajos futuros con este u otros conjuntos de datos.

Cominity Edge Gini Complement como acierto

Por último probamos a optimizar CEGC. En este caso las versiones optimizadas sobrepasan a sus respectivas versiones estándar. De nuevo *Thompson Sampling* vuelve a obtener el mejor resultado y además la penalización al acierto acumulado resulta despreciable. En lo que corresponde a *Epsilon Greedy* el descenso en acierto acumulado es algo mayor y al obtener básicamente los mismos resultados que con la versión habitual de este algoritmo puede considerarse que no compensa utilizar dicha versión optimizada.

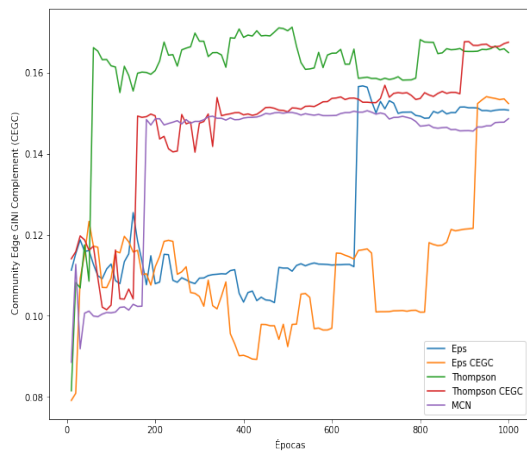


(a) Evolución de CCC en función del número de épocas

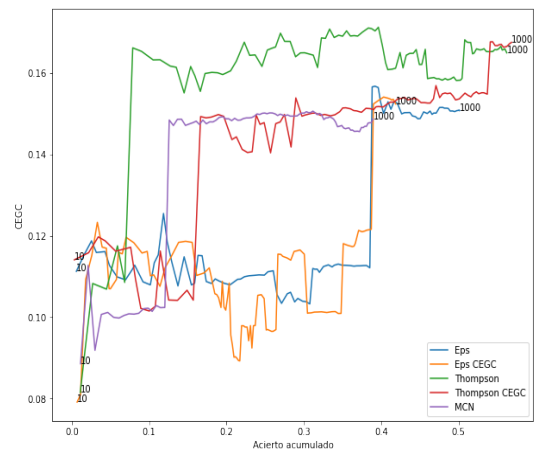


(b) Tradeoff entre Acuerdo acumulado y CCC

Figura 5.9: Resultados de utilizar CCC como acierto en los bandidos multi-brazo



(a) Evolución de CEGC en función del número de épocas



(b) Tradeoff entre Acuerdo acumulado y CEGC

Figura 5.10: Resultados de utilizar CEGC como acierto en los bandidos multi-brazo

CONCLUSIONES

El presente TFM ha estudiado la evolución de una red social al utilizar un recomendador de contactos. Focalizando en el estudio de ciertas métricas del grafo intrínseco de la red social y observando su evolución al aplicar diferentes algoritmos de recomendación. El estudio se ha llevado a cabo mediante la implementación de un modulo de recomendación de contactos con carácter cíclico y la implementación y adaptación de varios algoritmos. Con estas herramientas hemos realizado un análisis de las características de una subred real de Twitter cuyas conclusiones se exponen en el siguiente apartado.

6.1. Resumen y contribuciones

El primer experimento ha consistido en observar como evolucionan una serie de características de la red al utilizar diferentes algoritmos de recomendación. Concretamente se comparan 2 grandes enfoques de recomendación, la recomendación basada en vecindarios con la recomendación mediante bandidos multi-brazo. De este experimento resulta interesante ver como se adaptan a la perspectiva interactiva algoritmos clásicos de recomendación que no estaban diseñados con este enfoque y además comparar los resultados con algoritmos novedosos que están teniendo un fuerte impacto en el estado del arte.

En los resultados observados en la sección 5.2 se observa como en función de que métrica estemos observando conviene utilizar un algoritmo u otro. Por el contrario, existen otras métricas en los que según avanza el tiempo en la red social los resultados de los diferentes algoritmos se aproximan, esto indica la existencia de características intrínsecas en la red social.

El segundo experimento se limita a observar los cambios que se producen en los bandidos multi-brazo al considerar nuevas nociones de acierto. Los resultados son positivos, se observa como es posible obtener mejores resultados para las métricas al cambiar la noción de acierto de los bandidos. Destacamos que se pierde precisión al realizar estas optimizaciones sin embargo existen casos en los que el *tradeoff* métrica-precisión resulta interesante.

6.2. Trabajo futuro

A raíz de los resultados observados se plantean nuevas preguntas. La principal es si los resultados observados para estos algoritmos se repite para los demás paradigmas de recomendación. Por ello queda abierto repetir esta comparativa entre otros algoritmos.

Además este TFM muestra que resulta viable realizar variaciones en la noción de acierto del bandido multi-brazo, queda abierto investigar con nuevas estrategias de acierto y profundizar en el impacto que esto genera en las recomendaciones y en la propia red.

BIBLIOGRAFÍA

- [1] Lada A Adamic and Eytan Adar. Friends and neighbors on the web. *Social networks*, 25(3), 2003.
- [2] Charu C. Aggarwal. *Recommender Systems: The Textbook*. Springer International Publishing, 2016.
- [3] Robin Burke. Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, 12(4), 2002.
- [4] Jilin Chen, Werner Geyer, Casey Dugan, Michael Muller, and Ido Guy. Make new friends, but keep the old: recommending people on social networking sites. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2009.
- [5] Leo A. Goodman. Snowball sampling. *Ann. Math. Statist.*, 32(1):148–170, 03 1961.
- [6] P. Castells J. Sanz-Cruzado. Enhancing structural diversity in social networks by recommending weak ties. In *Enhancing Structural Diversity in Social Networks by Recommending Weak Ties*. ACM, 2018.
- [7] David Liben-Nowell and Jon Kleinberg. The link-prediction problem for social networks. *Journal of the American society for information science and technology*, 58(7), 2007.
- [8] Esther López Ramos et al. Bandidos multi-brazo en sistemas de recomendación. B.S. thesis, UAM, 2019.
- [9] Mark EJ Newman. Clustering and preferential attachment in growing networks. *Physical review E*, 64(2), 2001.
- [10] Javier Sanz-Cruzado and Pablo Castells. Contact recommendations in social networks. In *Collaborative Recommendations: Algorithms, Practical Challenges and Applications*. World Scientific Publishing, November 2018.
- [11] Tao Zhou, Linyuan Lü, and Yi-Cheng Zhang. Predicting missing links via local information. *The European Physical Journal B*, 71(4), 2009.

