# ESCUELA POLITÉCNICA NACIONAL

## FACULTAD DE INGENIERÍA EN SISTEMAS

## EVALUACIÓN DEL DESEMPEÑO COMPUTACIONAL DE SISTEMAS DE RECOMENDACIÓN APLICADO A BASES DE DATOS FARMACOLÓGICAS

### Sistema de recomendación basado en contenido

**TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN CIENCIAS DE LA COMPUTACIÓN**

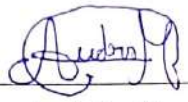
**CARLOS ANDRÉS MURGUEYTIO CAMPAÑA**

**DIRECTOR: IVÁN MARCELO CARRERA IZURIETA**

**DMQ, Julio 2022**

# CERTIFICACIONES

Yo, Carlos Andrés Murgueytio Campaña declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.



**Carlos Andrés Murgueytio Campaña**

Certifico que el presente trabajo de integración curricular fue desarrollado por Carlos Andrés Murgueytio Campaña, bajo mi supervisión.

**Iván Marcelo Carrera Izurieta**

**DIRECTOR**

# DECLARACIÓN DE AUTORÍA

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el (los) producto(s) resultante(s) del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

CARLOS ANDRÉS MURGUEYTIO CAMPAÑA

IVAN MARCELO CARRERA IZURIETA

# DEDICATORIA

Este Trabajo de Integración Curricular está dedicado especialmente a mi familia más cercana: mi madre Janeth Campaña, que es la persona más importante en mi vida a quien dedico y agradezco todos mis logros personales y profesionales, a mi tía Olga Campaña y mi tío Wilfrido Campaña, que han sido pilares fundamentales durante toda mi vida y han sido y seguirán siendo mis guías y una fuente de inspiración para mí. A mi primo Santiago Murgueytio, que considero mi hermano, que me ha acompañado sobre todo en los momentos más difíciles.

También quiero hacer una mención honorífica a mi abuelito Melchor Campaña, que durante toda su vida fue mi ejemplo a seguir, y quien sé que aún que no esté conmigo, seguirá acompañándome a donde quiera que vaya.

Agradecer finalmente a todas las personas que me han apoyado no solo en mi carrera universitaria, sino en todo aspecto de mi vida, a mis amigos más cercanos a quienes conozco desde niños, que son personas que valoro y quiero mucho, también a esos amigos que descubrí durante mi carrera universitaria y me han acompañado durante todos estos años, finalmente agradecer a todas las personas que han aportado su granito de arena en mi vida.

# AGRADECIMIENTO

Quiero agradecer especialmente a mi familia y amigos, muchas gracias por su apoyo, amistad y cariño, me enseñaron que el amor si existe, ninguno de mis logros sería posible sin ninguno de ustedes.

También quiero agradecer a la Escuela Politécnica Nacional y a todo su cuerpo docente y administrativo, que han sido claves en mi carrera profesional y a quienes les debo muchas enseñanzas.

Finalmente, quiero agradecer a mi director, el Ing. Iván Carrera, que me apoyó durante todo el tiempo que tomó realizar este Trabajo, muchas gracias por sus enseñanzas y su paciencia.

# ÍNDICE DE CONTENIDO

# ÍNDICE DE TABLAS



# ÍNDICE DE FIGURAS

# RESUMEN

En la actualidad, en el ámbito de la bioinformática, nuevos descubrimientos o hallazgos científicos hacen que surjan nuevos problemas o complicaciones sobre distintos temas, por ejemplo, cada cierto tiempo se actualiza la información sobre nuevos fármacos en la base de datos química ChEMBL, por lo que toda la información relacionada a esos nuevos fármacos debe ser procesada y verificada, lo que resulta en un trabajo que conlleva mucho esfuerzo y tiempo. Además, algunas veces este trabajo no es tan eficiente o no arroja resultados convincentes, lo que genera que estos descubrimientos y esfuerzos no sean relevantes. Frente a esta problemática, una de las soluciones que surgen es el reposicionamiento de fármacos.

El presente trabajo de titulación muestra el procedimiento para elaborar un sistema de recomendación basado en contenido, para realizar el reposicionamiento de los fármacos que existen en la versión 29 de ChEMBL, los resultados obtenidos con el sistema de recomendación serán comparados con los resultados reales, lo que ayudará a verificar la exactitud de las recomendaciones realizadas.

**PALABRAS CLAVE:** base de datos química, reposicionamiento de fármacos, sistema de recomendación, sistema de recomendación basado en contenido, generación de recomendaciones.

# ABSTRACT

Nowadays, in the field of bioinformatics, new scientific discoveries or findings cause new problems or complications to arise on different topics, for example, every so often information about new drugs is updated in the chemical database ChEMBL, so all the information related to these new drugs must be processed and verified, which results in a work that involves a lot of effort and time. In addition, sometimes this work is not as efficient or does not yield convincing results, which means that these discoveries and efforts are not relevant. Faced with this problem, one of the solutions that arise is drug repositioning.

This degree work shows the procedure to elaborate a content-based recommendation system to perform the repositioning of the drugs that exist in version 29 of ChEMBL. The results obtained with the recommendation system will be compared with the actual results, which will help to verify the accuracy of the recommendations made.

**KEYWORDS:** chemical database, drug repositioning, recommender system, content-based recommender system, recommendations generation.

# 1 DESCRIPCIÓN DEL COMPONENTE DESARROLLADO

Hoy en día existen infinidad de fármacos que, durante muchos años, han sido diseñados para un único fin, como cura de una única enfermedad o como tratamiento de una enfermedad concreta, pero puede haber casos en los que la eficacia de estos fármacos no ha sido lo suficientemente alta o las contraindicaciones que presentan son demasiado graves, por lo que han sido retirados del mercado. También puede haber casos en los que se han descubierto nuevos fármacos más eficaces, que han dejado obsoletos a los primeros.

Por otro lado, hay casos en los que se ha comprobado que un fármaco que fue creado para una afección específica ha funcionado para otra enfermedad que no está relacionada con la primera. A partir de esta problemática surge el reposicionamiento de fármacos cuyo objetivo es lograr que los medicamentos que se utilizan para una enfermedad específica o aquellos fármacos obsoletos tengan una nueva utilidad, de esta manera se logra ampliar el horizonte y las posibilidades que limitan un innumerable número de fármacos.

Uno de los beneficios que se pueden obtener con el reposicionamiento de medicamentos es aprovechar todo el tiempo y los recursos que existen detrás del desarrollo de cada fármaco. Toda la investigación que conlleva la creación de fármacos es inmensa, y a veces los investigadores tardan años en dar con una nueva fórmula, por lo que todo este trabajo se desperdicia o no se aprovecha todo su potencial. Para evitar este problema, el reposicionamiento juega un papel fundamental, reduciendo así su impacto.

Otro beneficio que podría generar el reposicionamiento de fármacos es en el campo de las enfermedades poco conocidas o cuyo tratamiento no es eficaz, bien porque el fármaco es demasiado caro o porque no representa una cura para la enfermedad.

Las técnicas que existen actualmente para el reposicionamiento de fármacos son numerosas, una de ellas trata sobre los sistemas de recomendación, en el presente documento, en específico, se va a trabajar en un sistema de recomendación basado en contenido.

## 1.1 Objetivo general

El presente trabajo de integración curricular tiene como objetivo general implementar y evaluar un sistema de recomendación basado en contenido aplicado a bases de datos farmacológicas con el fin de reposicionar fármacos obsoletos o cuyo uso es limitado.

## 1.2 Objetivos específicos

1. Definir la importancia de los sistemas de recomendación en el reposicionamiento de fármacos.
2. Extracción de información relevante de las bases de datos químicas.
3. Diseñar un sistema de recomendación basado en contenido para el reposicionamiento de fármacos.
4. Evaluar el sistema de recomendación con el fin de verificar su precisión.

## 1.3 Alcance

El objetivo final de este componente es desarrollar e implementar un sistema de recomendación basado en contenido aplicado a bases de datos farmacológicas con el fin de predecir las interacciones farmacológicas. Después se realizará una evaluación del desempeño computacional de dicho sistema, utilizando como métrica la precisión y el tiempo que se demora en realizar el procedimiento.

También se evaluará esta implementación mediante el error que exista entre la recomendación generada por el sistema con la similitud actual que existen entre los mismos en una versión más actual de la base de datos.

## 1.4 Marco teórico

### 1.4.1 Base de datos química ChEMBL

La base de datos química de acceso púbico, ChEMBL, es una base de datos "quimiogenómica" que reúne datos químicos, de bioactividad y genómicos para ayudar a traducir la información genómica en nuevos medicamentos eficaces. [1] Dentro de esta base se encuentra información que fue utilizada para este trabajo de integración curricular, por ejemplo, fármacos que se encuentran en estado de comercialización, líneas celulares que interactúan con estos fármacos y las interacciones que tiene un fármaco con una determinada línea celular, lo que genera el banco de datos principal que se utilizó.

Para realizar el presente trabajo, se utilizó esta base de datos en dos versiones, la primera que es la versión número 29, que fue utilizada para entrenar el sistema de recomendación y la versión número 30, que fue utilizada para generar las recomendaciones. Para evaluar estas recomendaciones se utilizó un método denominado “Evaluación por utilidad” que es explicado en el siguiente apartado.

### 1.4.2 Evaluación por utilidad

En la evaluación de un sistema de recomendación basada en la utilidad, la idea básica es que cada elemento del conjunto de datos valorado por un usuario tiene una utilidad para el usuario, que depende tanto de su posición en la lista recomendada como de su valoración real. Un elemento que tiene una calificación más alta tiene obviamente una mayor utilidad para el usuario. Además, los elementos mejor clasificados en la lista recomendada tienen una mayor utilidad para el usuario porque es más probable que se fijen en ellos (en virtud de su posición) y que acaben siendo seleccionados. Lo ideal sería que los artículos con una mayor puntuación se colocaran lo más alto posible en la lista de recomendaciones. [2]

### 1.4.3 SMILES de un compuesto

Los SMILES o por su notación en inglés Simplified Molecular Input Line Entry System de un compuesto es una notación de línea (un método tipográfico que utiliza caracteres imprimibles) para introducir y representar moléculas y reacciones [3]. Es decir, son representaciones estándar de las fórmulas químicas de los compuestos. Es un valor único, por lo que sirve para diferenciar a un compuesto de otro.

### 1.4.4 Fingerprint de un compuesto

Los fingerprint son una representación muy abstracta de ciertas características estructurales de una molécula o compuesto [4]. A grandes rasgos, el fingerprint es la representación binaria de un compuesto, que se obtiene a través del SMILES del mismo. Para el presente trabajo, se utilizó para medir la similitud entre compuestos.

### 1.4.5 Coeficiente de Tanimoto

El coeficiente de Tanimoto es la cantidad de elementos por la que dos individuos expresan cierta preferencia o afinidad, dividida por la cantidad de elementos por los cuales el usuario expresa cierta preferencia o afinidad [5]. En este trabajo, se utilizó el coeficiente de Tanimoto como métrica de similitud entre los fingerprint de un compuesto, para hallar la similaridad entre estos compuestos.

De manera adicional, se tomó en cuenta el estándar de similitud química establecido para la mayoría de las aplicaciones de la quimio informática, se consideró que un par de moléculas son estructuralmente similares si comparten un coeficiente de Tanimoto igual o superior a 0,55 [6].

### 1.4.6 Reposicionamiento de fármacos

En el ámbito de la bioinformática, el proceso de descubrir o desarrollar un nuevo fármaco es un esfuerzo que conlleva mucho tiempo y recursos, además de que algunas veces sucede que se crea un nuevo fármaco para una enfermedad o virus específico, pero los resultados que este genera no son lo suficientemente buenos o estables para que sea considerado útil o relevante, lo que genera que este tiempo y recursos invertidos sean desperdiciados, es entonces donde entra el reposicionamiento de fármacos, que en resumen es el proceso de encontrar una nueva indicación o uso para un fármaco que ya está aprobado o que ya cumplió las etapas iniciales para su aprobación. El reposicionamiento conlleva una reducción en los costes y el tiempo empleados [7].

### 1.4.7 Sistemas de recomendación

Los sistemas de recomendación son herramientas de análisis de datos de usuarios, en las que, de acuerdo con ciertas características de los usuarios, se generan recomendaciones de elementos que pueden ser de valor o utilidad para ellos. [8]. Los sistemas de recomendación cuentan con dos partes primordiales, la primera que es una entidad a la

que se proporciona la recomendación se denomina usuario, y el producto que se recomienda también puede ser denominado artículo. [2]. Pueden utilizar reseñas, valoraciones, o cualquier tipo de retroalimentación de los usuarios sobre un determinado artículo, para inferir los intereses del cliente, estos sistemas suelen basarse en la interacción previa entre el usuario y los artículos cuyas características o atributos son similares a los que van a ser recomendados. Pero este no es el único caso de sistema de recomendación, otro caso es cuando el usuario especifica sus preferencias, por lo que el sistema recomienda artículos según sus preferencias. Entonces, se podría decir que el objetivo principal de un sistema de recomendación es convertir los datos de los usuarios y sus preferencias en recomendaciones de los posibles gustos e intereses futuros de los usuarios. [9]

Para poder implementar un sistema de recomendación se utiliza un conjunto de datos cuyas características varían dependiendo el tipo de sistema de recomendación, de manera general se puede decir que necesitamos información relevante que el usuario genera al utilizar cualquier tipo de aplicación, es decir, los artículos que visita y/o compra, las reseñas que podría realizar sobre estos, puntuaciones, cuantas veces ha visitado ese artículo, etc.

### 1.4.8 Tipos de sistemas de recomendación

Existen varios tipos de sistemas de recomendación:

- Modelos colaborativos: Este tipo de sistema se refiere al uso de valoraciones de múltiples usuarios de forma colaborativa para predecir las valoraciones que faltan.

- Modelos basados en el conocimiento: En este tipo de sistemas los usuarios especifican de forma interactiva, y la especificación del usuario se combina con el conocimiento del dominio para proporcionar recomendaciones.

- Sistemas de recomendación demográfica: Este tipo de sistemas de recomendación utilizan la información demográfica del usuario, para categorizarlo y generar mejores recomendaciones. [2]

Existen más tipos de sistemas de recomendación, pero se va a realizar especial énfasis principalmente en el sistema de recomendación basado en el contenido, que básicamente es un sistema de recomendación que trabaja directamente con las interacciones que tiene un usuario con diferentes artículos, este tipo de sistema de recomendación es explicado a mayor profundidad en el siguiente apartado, pues es el modelo que ha sido implementado para el presente trabajo.

### 1.4.9 Sistemas de recomendación basados en el contenido

Los sistemas de recomendación basados en el contenido están diseñados para poder utilizar todas las características o atributos que cada usuario ha generado al interactuar con los artículos de su ambiente. Estas características son suficientes para descubrir recomendaciones significativas o relevantes para ese usuario. Esta perspectiva resulta sumamente útil cuando un artículo es relativamente nuevo y cuenta con pocas o casi ninguna interacción con otros usuarios. Uno de los objetivos principales de los sistemas de recomendación basados en el contenido es intentar generar un vínculo en el que los usuarios sean recomendados con nuevos artículos que compartan alguna característica con aquellos artículos que han sido de su agrado o han tenido interacción con anterioridad. Es importante destacar que los usuarios se encuentran aislados entre sí, es decir, las interacciones usuario – artículo no influyen en nada o casi nada sobre las recomendaciones realizadas sobre un usuario en concreto [2].

### 1.4.10 Citotoxicidad

La citotoxicidad se refiere a la capacidad de una molécula o compuesto a ocasionar algún tipo de daño celular [10]. Para este trabajo se utilizaron 4 métricas de citotoxicidad explicadas a continuación:

- IC50: se utiliza para determinar la concentración mínima de inhibición para inhibir el 50% de un patógeno. [11]
- CC50: representa la concentración citotóxica de los extractos para causar la muerte del 50% de las células viables del huésped. [11]
- EC50: es la concentración de un fármaco que da una respuesta semi-máxima. [12]
- GI50: es la concentración para el 50% de la inhibición máxima de la proliferación celular [13]

# 2 METODOLOGÍA

El procedimiento que se realizó durante todo el trabajo se describe a detalle a continuación.

Lo primero que se hizo fue acceder a la base de datos ChEMBL en su versión número 29 y se la descargó. De manera local se hizo el procesamiento y lectura de esta, después se obtuvieron las siguientes características de la sección perteneciente a los datos experimentales, como se puede ver en la Figura 1:

- El identificador de la actividad
- El identificador del ensayo
- El número del registro de cada molécula
- El identificador de la célula que interactúa con dicha molécula
- El valor estándar
- Las unidades del valor estándar
- El tipo de valor estándar

```
connection = sqlite3.connect('../chembl_29_sqlite (1)/chembl_29_sqlite/chembl_29.db')
cursor = connection.cursor()
query = "SELECT ACT.ACTIVITY_ID, ASY.ASSAY_ID, ACT.MOLREGNO , ASY.CELL_ID, ACT.STANDARD_VALUE,ACT.STANDARD_UNITS"
    ", ACT.STANDARD_TYPE " \
    " FROM ACTIVITIES ACT INNER JOIN ASSAYS ASY ON ACT.ASSAY_ID = ASY.ASSAY_ID" \
    " WHERE ASY.CELL_ID IS NOT NULL"
cursor.execute(query)
activity_dictionary = cursor.fetchall()
activity_columns =['activity_id','assay_id','molregno','cell_id','std_value','std_units','std_type']
activity_df = pd.DataFrame(activity_dictionary,activity_columns)
```

*Figura 1 Lectura y procesamiento de datos de ChEMBL29*

A continuación, se procedió a filtrar estos resultados, el criterio que se utilizó fue que los tipos de valor estándar deben ser de cualquiera de estos 4 tipos o valores de citotoxicidad: IC50, CC50, EC50, GI50 lo cual se puede visualizar en la Figura 2.

```
raw_act = activity_df[activity_df['std_type'].isin(['IC50','CC50','EC50','GI50'])]
```

*Figura 2 Filtrado de datos por su tipo de valor estándar*

Posteriormente se aplicó otro filtro y se estandarizó cada uno de estos registros a una unidad estándar en común, la unidad seleccionada fue uM (micro moles), Como se puede ver en la Figura 3.

```
raw_act = raw_act[raw_act['std_units'].isin(['nM', '10^4M', '/uM', "10'-11uM", "10'10uM", "10'8pM",
                                             "10'7pM", "10'6pM", "10'5pM", "10'-4nM", "10'6uM", "10'5uM", 'µM'])]
```

*Figura 3 Filtrado de datos por unidad de valor estándar*

Después se añadió una columna extra al conjunto de datos denominada ‘actividad’, para obtener esta columna la lógica aplicada es que si el valor estándar es menor o igual a 10 uM (micro moles), esa interacción se la considera como positiva, entonces se coloca el valor de 1, caso contrario se coloca el valor de -1 que significa que esa interacción es negativa. Se puede visualizar este paso en la Figura 4.

```
raw_act['actividad']= np.where(raw_act['std_value']<=10.0,1,-1)
```

*Figura 4 Obtención de datos para la columna 'actividad'*

Finalmente, para esta parte, lo siguiente fue realizar un agrupamiento entre pares compuestos – linea celular que presenten solo uno de los dos posibles valores de actividad, pues de esta manera se omiten datos vacíos o que podrían afectar a los resultados del sistema de recomendación, como se puede ver en la Figura 5.

```
act = pd.DataFrame(raw_act[['molregno','cell_id','active']])
act.to_csv('../csvs/summary30.csv',header=True,index=True)
act=pd.read_csv('../csvs/summary30.csv',index_col=0).drop_duplicates(keep='first')
df_count = act.groupby(by=['molregno','cell_id'])
index1 = pd.MultiIndex.from_arrays([act[col] for col in ['molregno', 'cell_id']])
index2 = df_count.index
summ_act=act.loc[index1.isin(index2)]
summ_act
```

*Figura 5 Agrupamiento entre pares compuestos - línea celular*

Hasta este punto existía un desbalanceo de datos, es decir, un 97,91% del conjunto de datos presentaba un valor de actividad compuesto – célula de -1 y apenas 2,08% presentaba un valor de +1, como se indica en la Figura 6, lo que en un futuro podría resultar como algo negativo para nuestro modelo. Para solucionar esto se aplicó un nuevo filtro al conjunto de datos, en este caso, el procedimiento que se realizó fue seleccionar solo aquellos compuestos que reportan actividades o inactividades dentro del universo, con una o más células, de esta manera el nuevo conjunto de datos con el que se construyó el modelo puede ser más preciso en sus recomendaciones, lo cual se puede ver en la Figura 7.

```
size = summ_act.shape[0]
print (len(summ_act[summ_act['active'].isin([1])])/size)
print (len(summ_act[summ_act['active'].isin([-1])])/size)
```

```
0.0208616956290995
0.9791383043709005
```

*Figura 6 Muestra del desbalanceo de datos*

```
comp_in_act = summary.groupby(by=['compound_id', 'active']).size().groupby(by=['compound_id'])
        .size()[summary.groupby(by=['compound_id', 'active']).size().groupby(by=['compound_id']).size()==2].index
b_summary = summary[summary['compound_id'].isin(comp_in_act)]
b_summary
```

*Figura 7 Aplicación del filtro para balanceo de datos*

El siguiente paso fue elegir, dentro de este universo de muestras previamente filtrado, el top 3 de compuestos más parecidos entre ellos, esto se lo realizó mediante la comparación de fingerprint entre compuestos, primero de cada SMILE se obtuvo el fingerprint, después ese fingerprint se lo comparó a los fingerprint de otros compuestos, esta comparación da como resultado un valor de coeficiente de tanimoto. Después, para obtener los tres compuestos más parecidos al compuesto inicial el criterio utilizado fue almacenar aquellos compuestos que presenten un valor de coeficiente de tanimoto menor o igual a 0.55. Como resultado de este proceso, se obtuvo el conjunto de datos con el top 3 de cada compuesto que se puede visualizar en la figura 8, que contiene lo siguiente:

- Identificador del compuesto
- Top 3 de los compuestos que más se parecen a ese compuesto

| | comp_id | top3 |
|---|---|---|
| **0** | 97 | [97, 312495, 2507043, 2511460] |
| **1** | 115 | [115, 835256] |
| **2** | 146 | [146, 210, 1795, 506559] |
| **3** | 147 | [147, 48536] |
| **4** | 148 | [148] |
| **...** | ... | ... |
| **38911** | 2537371 | [2537371] |
| **38912** | 2537372 | [70140, 1076140, 2537372] |
| **38913** | 2537374 | [2537374] |
| **38914** | 2537375 | [4398, 302932, 1163762, 2537375] |
| **38915** | 2537376 | [1275851, 2505521, 2537376] |

*Figura 8 Conjunto de datos con el top3 de cada compuesto*

El siguiente conjunto de datos que se obtuvo para este proyecto contiene los perfiles de afinidad de cada célula, lo que quiere decir esto es que se necesitó obtener todos los compuestos que interactúan con una de las células en el conjunto de datos. Para esto se utilizó el ya descrito conjunto de datos balanceado respecto a actividades o inactividades (ver Figura 7), lo que se hizo fue iterar el mismo respecto a cada célula, después buscar aquellos compuestos que tenían cualquier tipo de interacción (+1 o -1) y añadirlos a la lista o mejor conocido como perfil de afinidad de esa célula en concreto, también de manera adicional, se guardó cada valor de interacción del compuesto respecto a la célula. Este proceso se lo realizó para cada una de las células del conjunto de datos, como se puede visualizar en la Figura 9.

```
lista_compuestos = []
for i in b_summary['cell_id'].unique(): #i es el cell id
    lista_compuestos.append([i, list(b_summary[b_summary['cell_id' ]== i]['compound_id']),
                             list(b_summary[b_summary['cell_id' ]== i]['active'])])
```

*Figura 9 Obtención del perfil de afinidad para cada línea celular*

Como resultado se obtuvo un nuevo conjunto de datos con las siguientes características que se ve reflejado en la Figura 10:

- Identificador de la célula
- Lista de compuestos con los que interactúa (perfil de afinidad)
- Valor de la actividad de cada uno de esos compuestos con la célula

| | cell_id | id_compuestos_actividad | activity |
|---|---|---|---|
| **841** | 1 | [2580, 5105, 13590, 178357, 1148287] | [1, 1, -1, -1, 1] |
| **1190** | 2 | [42503, 444713, 1337017] | [-1, -1, -1] |
| **973** | 3 | [5769, 16758, 16949, 103459, 287231, 446279, 2... | [-1, -1, -1, -1, 1, -1, 1] |
| **567** | 5 | [717, 2208, 5105, 5268, 8633, 11305, 13590, 49... | [-1, -1, 1, -1, -1, -1, -1, -1, -1, 1, 1, -1, ... |
| **1236** | 7 | [68607, 1704201, 1704202] | [-1, -1, -1] |
| **...** | ... | ... | ... |
| **1342** | 5924 | [232036, 232036, 2469807, 2469807] | [1, -1, -1, 1] |
| **1394** | 5927 | [462435] | [-1] |
| **1157** | 5928 | [25906, 183934] | [-1, -1] |
| **1367** | 5930 | [368538] | [-1] |
| **747** | 5931 | [923, 2471419, 2471677, 2471958, 2473036, 2475... | [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1] |

*Figura 10 Perfil de afinidad para cada línea celular*

Lo siguiente que se realizó fue trabajar con la versión 30 de ChEMBL, pues el objetivo es predecir el valor de actividad para las interacciones compuesto – línea celular nuevas que aparecieron en dicha versión, primero se realizó un procedimiento similar al explicado previamente en este documento para la versión 29. Se realizó todo el procedimiento hasta llegar al punto donde se tuvo la relación compuesto – línea celular con la respectiva actividad registrada para cada relación, como referencia, ver la Figura 5.

Una vez obtenido este conjunto de datos, el siguiente paso fue separar aquellas relaciones compuesto – linea celular que sean nuevas, como se explicó en el párrafo anterior. Para esto se hizo una comparación entre los dos conjuntos de datos, el primero perteneciente a la versión 29 y el segundo a la versión 30, ambos pertenecientes a la base de datos de ChEMBL. Lo que se hizo fue obtener un conjunto diferencia que contenga cada relación compuesto – linea celular que esté en la versión 30 pero que no esté en la versión 29, de esta manera se obtuvieron las relaciones nuevas de la versión 30. De manera gráfica, se puede visualizar mejor este procedimiento en la Figura 11. Todo este procedimiento está detallado en la Figura 12.

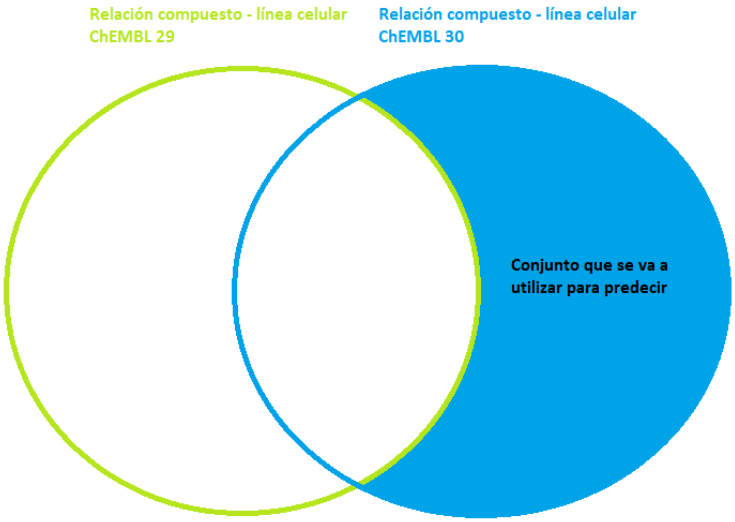



*Figura 11 Gráfico de ejemplo del conjunto diferencia entre versiones de ChEMBL*

```
index1 = pd.MultiIndex.from_arrays([v30[col] for col in ['compound_id', 'cell_id']])
index2 = pd.MultiIndex.from_arrays([v29[col] for col in ['compound_id', 'cell_id']])
v30_ni = v30.loc[~index1.isin(index2)]
v30_ni.columns =['compound_id','cell_id','active']
v30_ni
```

*Figura 12 Procedimiento para obtener el conjunto diferencia entre las dos versiones de ChEMBL*

Finalmente, para esta parte del proyecto, se incluyó una nueva columna a este conjunto diferencia, donde se va a guardar la actividad predicha por el sistema, como se puede ver en la Figura 13, con el fin de realizar una comparación una vez terminado el proceso.

| compound_id | cell_id | active | prediccion |
|---|---|---|---|
| 97 | 5673 | -1 | |
| 115 | 5673 | -1 | |
| 146 | 5673 | -1 | |
| 147 | 5673 | -1 | |
| 148 | 325 | -1 | |
| ... | ... | ... | ... |
| 2537375 | 5673 | -1 | |
| 2537376 | 303 | -1 | |
| 2537376 | 721 | -1 | |
| 2537376 | 786 | -1 | |
| 2537376 | 646 | -1 | |

*Figura 13 Conjunto de datos utilizados para el sistema*

Con todos los conjuntos de datos descritos previamente, ya es posible implementar el sistema de recomendación basado en contenido, los conjuntos de datos utilizados son:

- Conjunto de datos a predecir (Ver Figura 13).
- Top3 de los compuestos (Ver Figura 8)
- El perfil de afinidad de cada célula (Ver Figura 10)

Las funciones adicionales que se utilizaron en el algoritmo son las siguientes:

- Función que retorna el perfil de afinidad de una línea celular.
- Función que retorna el top 3 de un compuesto.

Ambas funciones pueden ser visualizadas en la Figura 14.

```python
#esta función retorna la lista de compuestos que interactuan con una celula, además de la actividad
def get_listaCompuestos_conActividad(cell):
    try:
        lista_compuestos_actividad =
        [eval(perfil_afinidad_activity['id_compuestos_actividad'][perfil_afinidad_activity['cell_id']==cell].tolist()[0]),
                        eval(perfil_afinidad_activity['activity'][perfil_afinidad_activity['cell_id']==cell].tolist()[0])]
    except:
        lista_compuestos_actividad = []
    return lista_compuestos_actividad
```

```python
#esta función retorna los compuestos (en forma de lista) pertenecientes al top3 del compuesto que se envia como parámetro
def get_top3_compound(compuesto):
    top3_compuesto = []
    try:
        top3_compuesto = eval(top3['top3'][top3['comp_id'] == compuesto].tolist()[0])
    except:
        pass
    return top3_compuesto
```

*Figura 14 Funciones utilizadas en el algoritmo del sistema*

Finalmente, el algoritmo utilizado es el siguiente. Primero, se obtiene cada compuesto y línea celular que se va a utilizar para la recomendación. Se obtiene el perfil de afinidad de esa línea celular. Después se itera cada uno de los compuestos del perfil de afinidad de esa línea celular, con la finalidad de obtener su top 3. Se procede a verificar que el compuesto perteneciente a la recomendación se encuentre dentro del top3, en caso de que efectivamente dicho compuesto esté dentro del top 3, se almacena de manera temporal el valor de la interacción que tiene la línea celular con ese compuesto del top 3.

Una vez se termine de iterar el perfil de afinidad, se procede a verificar los valores de interacciones obtenidos. Si todos los valores obtenidos son iguales, por ejemplo -1 o 1, la recomendación de ese par compuesto – línea celular tiene ese valor de -1 o 1. En caso de que haya una diferencia en los valores obtenidos, el valor de la recomendación se coloca como 0. Y, en caso de que no exista el compuesto en ningún top 3 o no se tenga un perfil de afinidad de una línea celular, no se puede realizar una recomendación, por lo tanto, se coloca el valor de ‘NA’ a la recomendación.

Este algoritmo puede ser visualizado en la Figura 15.

```
for i in conj_to_predict.index: #Recorro el conjunto de datos a predecir
    array_interaccion = []
    compuesto_a_predecir = conj_to_predict['compound_id'][i] #Par compuesto - línea celular al que se
    linea_celular_a_predecir = conj_to_predict['cell_id'][i] #le va a realizar la predicción
    perfil_afinidad = get_listaCompuestos_conActividad(linea_celular_a_predecir) #Obtención del perfil de afinidad
    if perfil_afinidad:
        for j in perfil_afinidad[0]:
            top3_cada_compuesto = get_top3_compound(j)#Obtención del top3 de cada compuesto del perfil de afinidad
            if top3_cada_compuesto:
                if compuesto_a_predecir in top3_cada_compuesto:
                    array_interaccion.append(perfil_afinidad[0].index(j))#Aquí se guarda el índice del valor de la interacción
    else:
        conj_to_predict.at[i, 'prediccion'] = 'NA' #En caso de que no exista un perfil de afinidad para la línea celular

    if len(array_interaccion) > 0:
        if len(array_interaccion) == 1: #Si existe solo un valor de interacción, se coloca ese valor en la predicción
            conj_to_predict.at[i, 'prediccion'] = perfil_afinidad[1][array_interaccion[0]]
        elif len(array_interaccion) > 1: #Se verifica que todos los valores de interacción sean iguales
            resultado = all(element == array_interaccion[0] for element in array_interaccion)
            if (resultado): #En caso de que todos los valores sean iguales
                conj_to_predict.at[i, 'prediccion'] = perfil_afinidad[1][array_interaccion[0]]
            else: #En caso de que no todos los valores sean iguales
                conj_to_predict.at[i, 'prediccion'] = 0
```

*Figura 15 Algoritmo perteneciente al sistema de recomendación*

# 3 PRUEBAS, RESULTADOS, CONCLUSIONES Y RECOMENDACIONES

## 3.1 Resultados

Una vez ejecutado el sistema de recomendación, se obtuvieron un total de 584 recomendaciones, tanto negativas con valor -1, positivas con valor 1 y otras con valor de 0.

Con la finalidad de interpretar de mejor manera estos datos, se realizó una matriz de confusión, la misma que puede ser visualizada en la Tabla 1.

*Tabla 1 Matriz de confusión*

| Valor recomendación/ Valor real | -1 | 0 | 1 |
|---|---|---|---|
| **-1** | 329 | 42 | 150 |
| **0** | 0 | 0 | 0 |
| **1** | 12 | 9 | 42 |

Como se puede visualizar en la matriz de confusión, el número de casos verdaderos positivos en caso de que una interacción sea negativa es de 329 y en caso de que una interacción sea positiva es de 42. Se tiene un total de 371 casos de verdaderos positivos.

En cuanto a falsos positivos se tiene lo siguiente, en caso de que el valor real sea -1 y el valor de la recomendación sea 0 se tienen 42 casos y para el valor de recomendación 1 se tienen 150 casos. Para los valores cuyo valor real es 1, los casos cuya recomendación fue de 0 son 9 y en los casos cuya recomendación fue de -1 son 12. Por lo que se obtiene un total de 213 falsos positivos.

También se calculó la precisión y la exhaustividad del sistema, como se puede visualizar en la Tabla 2.

*Tabla 2 Precisión y exhaustividad del sistema*

| Precisión y exhaustividad | Precisión | Exhaustividad |
|---|---|---|
| **Valor de -1** | 0,96 | 0,63 |
| **Valor de 0** | 0 | 0 |
| **Valor de 1** | 0,22 | 0,67 |

## 3.2 Conclusiones

- Se logró implementar un sistema de recomendación basado en contenido para las interacciones compuesto – línea celular de la base de datos ChEMBL, con un porcentaje de precisión del 96% para las interacciones negativas y del 22% para las interacciones positivas.
- Hay casos donde el sistema no puede realizar una recomendación, esto se debe a la falta de información que se tiene sobre un compuesto, por ejemplo, cuáles son las líneas celular con las que ha presentado una interacción. Por lo que lo más favorable para no alterar los resultados es ignorar o no realizar la recomendación sobre este compuesto.
- Los valores de 0 que se obtuvieron son para aquellas interacciones donde no hubo un consenso en el sistema de recomendación, es decir, aquellos compuestos cuyos compuestos más similares muestran una actividad tanto positiva como negativa, esto sin embargo no quiere decir que no se pueda realizar una recomendación para este tipo de compuestos.
- Si bien el porcentaje de precisión en cuanto a interacciones negativas es muy alto, para las interacciones positivas es muy bajo, esto puede ser debido a que la mayoría de las interacciones que se tiene del conjunto de datos son negativas, lo que genera una cierta inclinación a arrojar este tipo de recomendaciones.
- Se puede utilizar este sistema de recomendación para versiones futuras de la base de datos ChEMBL, siempre y cuando se tomen en cuenta los resultados obtenidos en este trabajo.

## 3.3 Recomendaciones

- Se recomienda recaudar información sobre las interacciones entre compuestos – línea celular de manera más exhaustiva, también sobre los tipos de valores estándar, así como las unidades de valor estándar, con el fin de generar nuevas recomendaciones y posibles nuevos resultados.

- Se recomienda investigar más sobre los sistemas de recomendación basados en el contenido, pues el presente trabajo de integración curricular muestra una primera aproximación a los mismos.

- Se recomienda realizar nuevas recomendaciones con otras versiones de la base de datos ChEMBL a manera de ejercicio práctico.

# 4 REFERENCIAS BIBLIOGRÁFICAS


[1] "What is ChEMBL? | ChEMBL", *Ebi.ac.uk*. [En Línea]. Disponible: https://www.ebi.ac.uk/training/online/courses/chembl-quick-tour/what-is-chembl/. [Último acceso: 13- agosto- 2022].

[2] C. Aggarwal, *Recommender Systems*. Cham: Springer International Publishing, 2016.

[3] “Daylight Theory: SMILES", Daylight.com. [En Línea]. Disponible: https://www.daylight.com/dayhtml/doc/theory/theory.smiles.html. [Último acceso: 20- agosto - 2022].

[4] "Daylight Theory: Fingerprints", Daylight.com. [En Línea]. Disponible: https://www.daylight.com/dayhtml/doc/theory/theory.finger.html. [Último acceso: 20- agosto - 2022].

[5] G. Mendoza Olguín, Y. Laureano de Jesús and M. Pérez de Celis Herrero, "Métricas de similaridad y evaluación para sistemas de recomendación de filtrado colaborativo", *Revista de Investigación en Tecnologías de la Información*, vol. 7, no. 14, 2019. Disponible: 10.36825/riti.07.14.019

[6] E. Tejera et al., "Cell fishing: A similarity based approach and machine learning strategy for multiple cell lines-compound sensitivity prediction", PLOS ONE, vol. 14, no. 10, p. e0223276, 2019. Disponible: 10.1371/journal.pone.0223276.

[7] M. Farha and E. Brown, "Drug repurposing for antimicrobial discovery", *Nature Microbiology*, vol. 4, no. 4, 2019. Disponible: 10.1038/s41564-019-0357-1.

[8] "Sistemas de recomendación | Qué son, tipos y ejemplos", *GraphEverywhere*. [En Línea]. Disponible: https://www.grapheverywhere.com/sistemas-de-recomendacion-que-son-tipos-y-ejemplos/. [Último acceso: 20- agosto - 2022].

[9] L. Lü, M. Medo, C. Yeung, Y. Zhang, Z. Zhang, and T. Zhou, "Recommender systems", *Physics Reports*, vol. 519, no. 1, pp. 1-49, 2012. Disponible: 10.1016/j.physrep.2012.02.006.

[10] "Potencial citotóxico | Océanos y salud humana", Oceanshealth.udg.edu. [En Línea]. Disponible: http://www.oceanshealth.udg.edu/es/potencial-citotoxico.html. [Último acceso: 20- agosto - 2022].

[11] J. van Rooyen, 2015. [En Línea]. Disponible: https://www.researchgate.net/post/Why-do-we-have-to-measure-IC50-CC50-and-SI. [Último acceso: 20- agosto - 2022].

[12] "50% of what? How exactly are IC50 and EC50 defined? - FAQ 1356 - GraphPad", Graphpad.com, 2010. [En Línea]. Disponible: https://www.graphpad.com/support/faq/50-of-what-how-exactly-are-ic50-and-ec50-defined/. [Último acceso: 20- agosto - 2022].

[13] S.N, 2020. [En Línea]. Disponible: https://www.researchgate.net/post/What-is-the-different-between-IC50-GI50-and-ED50. [Último acceso: 20- agosto - 2022].

# 5 ANEXOS

## ANEXO I

Repositorio con todo el procedimiento y conjunto de datos del presente Trabajo de Integración Curricular: https://github.com/Reyum/Sistema-de-Recomendaci-n-Basado-en-Contenido.git