Biopatrones: Un entorno lógico y ontológico para la exploración de conocimiento en redes de regulación genética.

GUÍA DEL INVESTIGADOR

Descripción, modos de consulta, análisis y resultados tipo.

(Versión 2.0-revisión-01)

López José[1,2], Jacinto Dávila[2] , Ramírez Yacson[1]

LCAR: Laboratorio de Computación de Alto Rendimiento, Universidad Nacional Experimental del Táchira, San Cristóbal, Táchira, Venezuela. jlopez@unet.edu.ve, yacson.ramirez@gmail.com .

CESIMO: Centro de Simulación y Modelos, Fac. Ingeniería. Universidad de Los Andes, Mérida, Venezuela. jacinto@ula.ve.

Tabla de contenidos:

Introducción.

Las redes de regulación genética (RRG) constituyen sistemas muy complejos con conjuntos de objetos biológicos muy diversos. Para realizar el modelado de tales redes, la comunidad científica se apoya en servicios informáticos disponibles a través de portales disponibles en Internet. Algunos de estos portales son: GeneOntology, PDB, HGNC, Pathway Commons, UniProt, PubMed, entre otros. Los sitios mencionados proporcionan servicios que hacen posible accederlos automáticamente y por lo tanto, estos pueden usarse para organizar bases de conocimiento que integren sus recursos. Nuestro equipo ha desarrollado el prototipo de un sistema que permite tal integración y el análisis de la información obtenida en diferentes modalidades. Servicios como el modelado automático de la identidad (p.e. receptor, enzima, etc),

funciones moleculares y procesos procesos biológicos de los objetos en una red y sus interacciones proteína-proteína, han sido implementados en la propuesta que aquí describimos. Exponemos una alternativa para el descubrimiento de patrones de señalización biológica y subredes regulatorias dentro de una RRG. Nuestro objetivo es expandir y mejorar las herramientas del sistema aquí descrito y sus posibles resultados, en la medida en que estas sean puestas a prueba por usuarios especializados en el tema. Este documento expone capacidades y modos de uso, que esperamos faciliten la definición de ajustes y nuevos requerimientos, orientados a mejorar la utilidad del sistema al que hemos llamado *biopatrones*. Para tal fin se emplea un ejemplo de modelado y análisis, en el que se desea explorar posibles vínculos entre los procesos regulatorios inherentes al SARS-COV y al SARS-COV-2. Lo aquí expuesto puede experimentarse directamente si se desea, pues se ha dispuesto un servidor para dar acceso a los investigadores interesados. Por lo pronto, el ejemplo mencionado solo ilustra el uso del sistema por lo que sus resultados sólo tienen un propósito académico. Este documento incluye dos secciones. La primera sección se enfoca en la interacción que típicamente el investigador realiza con el sistema. La segunda sección muestra lo requerido para ejecutar un experimento de modelado en el contexto de alguna red, más una traza de su ejecución, que describe todos los recursos y bases de conocimiento que el sistema organiza automáticamente. Se verá que el usuario puede con su intervención mejorar los resultados que el sistema le provee. Este documento incluye dos anexos. El primero describe cómo se recolecta y organiza la información que el sistema emplea y el segundo, que permite conocer detalles acerca de su descarga e instalación. Descargue el material complementario disponible en esta dirección, para asistir la descripción que aquí se realiza sobre los productos que el sistema genera. A continuación una descripción del contenido de esta guía.. Al final de la sección 2.1.1 (pág 14), UD ya estará corriendo un experimento.

Sección 1. **Modelando una RRG. Caso SARS-COV y SARS-COV-2** (páginas. 3 - 9).

Describe los menús principales del sistema y los resultados del experimento que el usuario replicará más adelante. Útil para tener una idea general de lo que el sistema ofrece.

Sección 2. **Biopatrones. Ejecutando un experimento.**

Sección 2.1. **Traza de ejecución y bases de conocimiento generadas.**

> Sección 2.1.1. Se detallan los pasos a seguir para replicar el experimento que se describe en la sección 1 (páginas. 9 - 14).
>
> Secciones 2.1.2 y 2.1.3. Ajustando bases de conocimiento (páginas. 14-17).
> Describe cómo puede el usuario ajustar el conocimiento automáticamente organizado, a fin de mejorar los resultados que el sistema le ofrece.
>
> Sección 2.1.4. Productos asociados a un experimento (páginas. 21 - 22).
> Describe las distintas etapas por las que pasa el sistema al colectar la información desde Internet y las bases de conocimiento que ello genera. Dado que un experimento puede durar días, la traza mantiene al usuario informado sobre la etapa en que se encuentra su experimento.

Sección 2.2. **El rol de las ontologías en biopatrones y cómo aprovecharlas.**(páginas. 20).

Útil para comprender el rol de las ontologías en el sistema y cómo aprovecharlas, a fin de lograr experimentos más ajustados a los intereses del investigador.

Sección 2.3. **Cómo podría el usuario escalar el experimento aquí descrito.**

El sistema provee diversos recursos que pueden guiar la ampliación del alcance de un experimento. Esta sección da algunos ejemplos de ello.

Anexo A. **Biopatrones. Sobre cómo se recolecta la información.**

Útil para conocer las fuentes consultadas a la Internet y proponer nuevas funcionalidades al sistema que las aprovechen.

Anexo B. **Biopatrones: Instalación y ejecución.**

Este anexo guía la instalación del sistema y describe sus componentes; además del modo de ejecutarlo. Se guía al usuario Windows sobre las herramientas a usar para acceder al servidor.

1. Modelando una RRG. Caso SARS-COV y SARS-COV-2.

Esta sección describe cómo modelar una red de regulación genética (RRG), dada una colección inicial de objetos biológicos sugeridos para esta. Se usa como ejemplo un experimento en el que se desea explorar posibles patrones y subredes, que sugieran algún tipo de relación entre el SARS-COV y el SARS-COV-2. La Tabla 1.1 muestra un extracto de la BC que se obtiene en este experimento (ver mc/minery/networks/COVID19/COVID19-IMMUNOLOGY/kBase.pl). Además de la BC mencionada, el sistema provee una versión documentada de la misma, que permite verificar la validez de los eventos regulatorios automáticamente modelados (ver kBaseDoc.txt en la misma ubicación). El par de BC mencionadas se complementan con otras demás bases de conocimiento, lo que permite dar forma a criterios según los cuales explorar la red; por ejemplo, explorar objetos que den forma a patrones (pathways) que cierren con algún objeto de interés (SARS-COV-2, por ejemplo).

1.1. Sobre la ejecución de biopatrones y la búsqueda de información.

En biopatrones, la ejecución de un experimento requiere información inicial que consiste en: 1) un listado de objetos biológicos inherentes a la red, en este caso objetos pertenecientes al virus mencionado definidos en el archivo expert_objects.txt (ver Tabla 1.2 y mc/data/COVID-19/COVID-19-IMMUNOLOGY/); 2) la posible región de regulación a estudiar (región UTR del SARS-COV-2, archivo covid19RegProm); 3) un listado opcional de objetos complementarios; por ejemplo, proteínas homólogas (no empleado en este caso); y 4) un listado opcional de PubMed IDs que el usuario desee sean considerados (archivo pubmed_IDsExp). Más adelante se verá cómo se proveen estos datos al sistema. Asuma que el sistema ya fue ejecutado y ha generado salidas. A continuación algunos detalles sobre el funcionamiento del sistema, el proceso de modelado y el tipo de análisis posibles, acompañado de algunos detalles sobre de las salidas que el sistema provee.

Según la Figura 1 el sistema organiza búsquedas por niveles. Para cada grupo de objetos definidos en un nivel de modelado, el sistema procede a realizar consultas tanto al Protein Data Bank, como a los servicios de identificación de objetos disponibles a través del HGNC. En el caso de PDB, este servicio responde informando sobre complejos que contienen a aquel objeto sobre el cual se consulta. Dado que PDB responde con complejos, el análisis de éstos permite obtener nuevos objetos posibles para la red; objetos que pasan a ser los nuevos objetos del siguiente nivel de construcción de la RRG (ver Figura 1). Por otro lado, HGNC provee metadatos acerca de cada objeto consultado, lo que incluye nombre oficial y sinónimos relacionados. Ambas consultas (PDB y HGNC) ofrecen la posibilidad de que la lista de objetos proporcionados inicialmente, sea extendida según criterios definidos por el usuario. La opción recién descrita sólo permite por lo pronto obtener complejos según el orden de relevancia que define el propio PDB. Se espera que los biólogos/médicos propongan otros criterios para explotar mejor esta funcionalidad. Para efectos de este ejemplo no se emplea la consulta a PDB pero es

una funcionalidad que el usuario podría explotar aportando sus propios criterios de búsqueda. Otros portales son también consultados para indagar información relativa a los objetos de una red en construcción, tal es el caso de GeneOntology y MeSH. Mayores detalles respecto a la descripción del sistema en el paper accesible en este enlace. El anexo A ofrece una introducción al respecto.

Tabla 1.1. Base de conocimiento parcial para ejemplo SARS-COV/SARS-COV-2.

```
base([
...
event('MHC',reveal,'PSMB7'),
event('MHC',develop,'PSMB7'),
event('STAT',relate,'CCR5'),
event('CCR5',relate,'STAT'),
event('CCR5',lead,'CCR5'),
event('Interferon',associate,'CCR5'),
event('STAT',activate,'CCR5'),
event('CCR5',activate,'STAT'),
….
event('ORF8',reveal,'SARS-CoV'),
event('SARS-CoV',reveal,'ORF8'),
event('SARS-CoV-2',associate,'ORF8'),
event('ORF8',associate,'SARS-CoV-2'),
event('SARS-CoV',associate,'ORF8'),
event('ORF8',emerge,'SARS-CoV-2'),
event('ORF8',emerge,'SARS-CoV'),
...
]).
```

Nota: La BC se modeló automáticamente y contiene 3908 eventos en total. Debe tenerse presente que al ser una BC automáticamente modelada esta presentará falsos positivos.

La Figuras 1 muestra que la BC modelada pasa por procesos automáticos de análisis, lo que eventualmente conduce a posibles patrones/pathways/caminos de regulación del estilo presentado en la Tabla 1.3. La Tabla 1.3 ilustra el resultado en la búsqueda de patrones presentes en la BC modelada (ver Tabla 1.1 y archivo pathways.txt, mc/minery/networks/COVID19/COVID19-IMMUNOLOGY/). La Tabla 1.3 muestra uno de varios patrones de regulación posibles. La misma tabla describe las oraciones que sustentan al patrón descubierto. Por otro lado, el sistema provee la versión documentada de la BC de eventos, lo que permite ubicar tales oraciones en el resumen general de los abstracts descargados por el sistema. Usando esas oraciones, puede uno dirigirse a PubMed y recuperar los PubMed IDs correspondientes a los eventos de un patrón de regulación.

La Tabla 1.3 incluye sólo uno de los patrones posibles, relativos a la posible relación de los objetos ORF6, CD4, JAK3, STAT y MHC en la regulación del SARS-CoV-2. Los patrones hallados deben ser analizados manualmente para determinar cuáles de ellos contienen solo eventos positivos. Se habla de eventos positivos, en el sentido de que la oración considerada resulta ser bien representada por el evento del caso. Para el ejemplo aquí presentado, el sistema arroja 401 patrones posibles, de los cuáles hemos elegido uno al azar. Dado que muchos patrones poseen al menos un evento falso positivo, estos deben considerarse errados; sin embargo, puede suceder que al leer la cadena de eventos del caso, el usuario descubra en ellos aquello que lo guié hacia algo de interés. Se verá más adelante que en escenarios como este, el usuario puede corregir o

agregar información y luego proceder a re-ejecutar al sistema en alguna etapa de su interés, mejorando así la calidad de sus resultados.

Tabla 1.2. Identificadores iniciales para SARS-COV/SARS-COV-2.

JAK
JAK3
MHC
ORF6
ORF8
importin
STAT
ACE2
CD4+
CD4
CD8+
CD8
CCR5
CXCR4
SARS-CoV-2
SARS-CoV
MICA
MICB
MICC
HLA-A
HLA-B
HLA-C

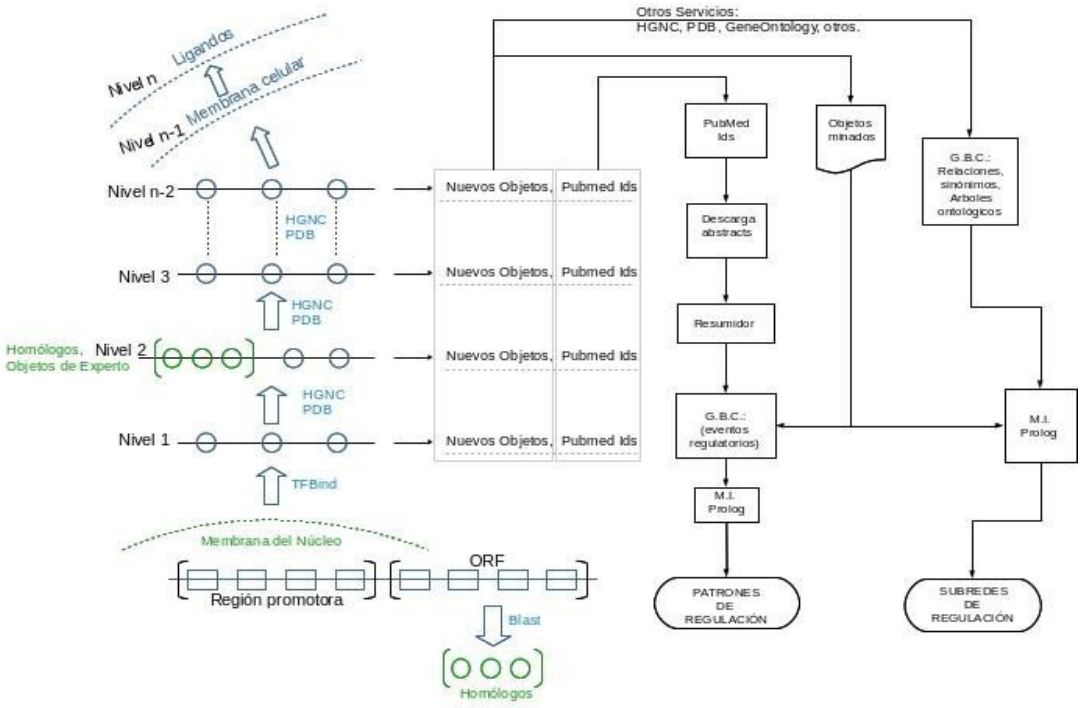

Figura 1. Arquitectura del sistema y dinámica para el modelado semántico y el análisis de una RRG.

1.2. Opciones de análisis.

Biopatrones está disponible para ser descargado e instalado, así como para ser instalado ejecutado remotamente en un servidor (detalles en el Anexo B). La Tabla 1.4 muestra un resumen de la navegación que típicamente se realiza al ejecutar el sistema. Como puede verse, primero se listan las redes en proceso de modelado. Una vez se ha elegido alguna red, se listan los distintos procesos que se han ejecutado o se están ejecutando para ella. Tan pronto como se ha elegido alguno de los procesos listados, el sistema presenta la configuración inicial que se asignó a dicho proceso, acompañado este de un resumen de los resultados obtenidos, tanto en lo relativo al modelado como a la búsqueda de patrones. Finalmente, el sistema despliega las opciones de análisis que provee para facilitar el estudio de la información, organizada para el proceso de modelado en consideración.

Tabla 1.3. Inferencia de patrones relativas a la regulación de SARS-CoV-2 y STAT (un ejemplo de 401 posibles).

P = 'ORF6',bind,'STAT';'STAT',associate,'JAK3';'JAK3',associate,'MHC';'MHC',associate,'CD4';'CD4',regulate,'SARS-CoV-2'

----> event: 'ORF6',bind,'STAT'

We mapped the region of ORF6, which binds karyopherin alpha 2 , to the C terminus of ORF6 and show that mutations in the C terminus no longer bind karyopherin alpha 2 or block the nuclear import of STAT1.

We also show that N-terminal deletions of karyopherin alpha 2 that no longer bind to karyopherin beta 1 still retain ORF6 binding activity but no longer block STAT1 nuclear import.

----> event: 'STAT',associate,'JAK3'

These results imply that JAK/STAT activation is associated with replication of leukemic cells and that therapeutic approaches aimed at JAK/STAT inhibition may be considered to halt neoplastic growth.

Multiple cytokine receptors signal through Janus kinases (JAKs) and associated signal transducer and activators of transcription (STATs).

----> event: 'JAK3',associate,'MHC'

These include HLA-B27 and the aminopeptidases (ERAP1, ERAP2, and LNPEPS) , which are involved in antigen processing and presentation to T-cells, and several genes (IL23R, IL6R, STAT3, JAK2, IL1R1/2, IL12B, and IL7R) involved in IL23 driven pathways of inflammation.

Compared with MHCC-97H-ROCK2, the DEmRNAs in MHCC-97H-ROCK1 were involved in the JAK-STAT cascade, the Akt signaling pathway and the activity of several different peptidases.

----> event: 'MHC',associate,'CD4'

In contrast, the increase in HLA-ABC expression by CD8+ lymphocytes was associated with transition from 2 H4+ to 2 H4int status, which suggests that increased HLA-ABC expression occurs at an earlier stage in the acquisition of CD45RO in CD8+ cells than for CD4+ cells.

After treatment, we found that the upregulation of PD-1 and T cell immunoglobulin mucin-3 (Tim-3) expression on CD4+ and CD8+ T cells was significantly associated with a poor clinical outcome in the HLA-A,2402-matched group (p = 0.033 , 0.0282 , 0.0046 , and 0.0068 , respectively).

----> event: 'CD4',regulate,'SARS-CoV-2'

Importantly, using this system, we functionally identified the CD4+ and CD8+ peptide epitopes targeted during SARS-CoV-2 infection in H2b restricted mice.

CD4+ T\240\cell responses to spike, the main target of most vaccine efforts, were robust and correlated with the magnitude of the anti-SARS-CoV-2 IgG and IgA titers.

|**Nota**: La documentación de cualquier patrón/pathway/camino puede accederse consultando el archivo *pathwaysDoc.txt* (ubicado en mc/minery/networks/COVID19/COVID19-IMMUNOLOGY/. Para consultarlo basta buscar allí el camino de interés.

La Tabla 1.4 muestra que para el experimento COVID-19-IMMUNOLOGY-I se analizaron 39 objetos. Puede verse además la cantidad de combinaciones de los identificadores de los objetos (nombres y sinónimos), según los cuales se realizaron las

consultas a PubMed; lo que condujo a 76881 PubMed IDs para el experimento en curso. Una vez descargados los abstracts del caso y realizado el proceso de identificación de oraciones regulatorias, el sistema construye automáticamente la BC mostrada en la Tabla 1.1. La Tabla 1.4 informa que para este experimento se modelaron 3908 eventos regulatorios.

Una vez que un experimento ha finalizado su fase de modelado, el sistema propone la exploración de patrones de regulación. La Tabla 1.4 muestra que tal paso ya se ejecutó al menos una vez, indicando 401 patrones disponibles. Concluida la búsqueda de patrones, queda disponible el menú de opciones para iniciar el análisis de la red (ver Tabla 1.4, parte baja). Destacan allí, en orden: 1) la posibilidad de ejecutar el modelado nuevamente; 2) acceder al menú general de análisis; 3) inferir nuevamente los patrones de regulación; y 4) repetir el proceso de modelado desde alguna etapa en particular (reconstruir la BC, por ejemplo). Esta última opción facilita que el usuario modifique cualquiera de las bases de conocimiento, solicitando posteriormente que se repitan etapas de modelado y análisis que sean de su interés.

Tabla 1.4. Un resumen de las opciones iniciales de biopatrones.

```
Seleccione una opción:
==== Redes existentes ====
1.- BAXS
2.- COVID-19
I.-  Ir a redes integradas
0.- Salida
2

Red:  COVID-19
Seleccione una opción:
==== Procesos existentes ====
1.- COVID-19
2.- COVID-19-Fármacos
3.- COVID-19-IMMUNOLOGY
4.- COVID-19-IMMUNOLOGY-I

I.-Hacer integración de la red
0.- Volver
4

biopatrones
minery/networks/COVID-19/COVID-19-IMMUNOLOGY

Configuración inicial:

*Región promotora:              COVID-19RegProm
*Cantidad de complejos:         0
*Número de niveles:             0
*Confiabilidad TFBind:          99
*Cantidad máxima de pubmed IDs:  2000
*Archivo de objetos del Experto: data/COVID-19/COVID-19-IMMUNOLOGY/pubmed_IDsExp

Resultados:

Objetos minados:           39
Combinaciones realizadas:   10521
Pubmed Id encontrados:      76881
Eventos encontrados:        3908
Patrones encontrados:       401menciosemenciose

Seleccione una opción:
1.- Crear un nuevo proceso.
2.- Ir al menú análisis de RRG.
3.- Inferir patrones.
4.- Actualizar la base de conocimiento de eventos.
5.- Documentar patrones de regulación.
6.- Reanudar desde.
0.- Volver
```

Tres opciones de análisis se ilustran en esta sección: 1) la búsqueda general de patrones de regulación presentes en la BC de eventos (ver tipo de resultados en la Tabla 1.3); 2) dado un ligando y un receptor, determinar el tipo de patrones a los que podrían conducir (inhibitorios o estimulatorios) (ver Tabla 1.5); y 3) la búsqueda de patrones que primero estimulan la presencia de un objeto en la red, conectados a otros patrones que luego pudieran inhibirla (ver Tabla 1.6). Las Tablas mencionadas son representativas del tipo de salidas que el sistema arroja y solo muestran parte del contenido correspondiente. Cada uno de los patrones presentados en tales tablas requiere de la validación del usuario de sus eventos regulatorios. Por lo pronto el sistema propone eventos, patrones y subredes, según restricciones biológicas que pueden ser ampliadas a solicitud del usuario. Al considerar los ejemplos, debe tenerse presente que dado que el modelado de eventos es automático, es normal ver falsos positivos tanto en los patrones como en los eventos de enlace.

Como se mencionó anteriormente, el sistema provee la posibilidad de reactivar cualquiera de sus etapas de modelado y ello es posible mediante la opción 6, visible en la parte baja de la Tabla 1.4. La Tabla 1.7 muestra el resultado de acceder a tal opción. El poder reactivar etapas específicas del modelado permite la participación activa del usuario en la mejora de resultados. Dado que cualquiera de las bases de conocimiento puede ser ajustada manualmente, tal opción permite afectar los resultados de etapas posteriores al caso que corresponda. Por ejemplo, supóngase que el sistema ya terminó de organizar las bases de conocimiento y que incluso ya se han inferido patrones. Supóngase además, que al revisar las oraciones de algún patrón de interés, se hizo evidente la importancia del objeto no considerado al principio del modelado. En este escenario el usuario puede agregar el nuevo identificador al listado de nombres y sinónimos; además de incluir manualmente la identidad del objeto en cuestión donde corresponda. Realizado esto se puede proceder a re-ejecutar la etapa 6 del sistema (Tabla 1.7), redefiniendo así la base de conocimiento relativa a eventos de regulación; lo que posiblemente sumará nuevos eventos en los que participe el objeto indicado. El escenario recién descrito es uno donde el usuario desea aprovechar la colección de abstracts que ya recolectó, mientras quizas ya corre un experimento desde cero, en el que el nuevo objeto ya ha sido incorporado.

Tabla 1.5. Tipo de patrones disponibles para un ligando/receptor dados (caso ORF6 / STAT).

'ORF6',bind,'STAT';'STAT',associate,'JAK3';'JAK3',associate,'MHC';'MHC',associate,'CD8';'CD8',regulate,'SARS-CoV-2'
['ORF6', 'STAT', 'JAK3', 'MHC', 'CD8', 'SARS-CoV-2']
Tipo: Estimulatorio

'ORF6',bind,'STAT';'STAT',associate,'JAK3';'JAK3',associate,'MHC';'MHC',associate,'CD8';'CD8',inhibit,'SARS-CoV-2'
['ORF6', 'STAT', 'JAK3', 'MHC', 'CD8', 'SARS-CoV-2']
Tipo: Inhibitorio

'ORF6',bind,'STAT';'STAT',associate,'JAK3';'JAK3',associate,'MHC';'MHC',associate,'CD8';'CD8',bind,'SARS-CoV-2'
['ORF6', 'STAT', 'JAK3', 'MHC', 'CD8', 'SARS-CoV-2']
Tipo: Estimulatorio

|**Nota**: En este ejemplo se proponen distintos patrones que podrían conducir a la estimulación o inhibición de la regulación del SARS-CoV-2. Se acuerda que ORF6 sea un ligando para el sistema, a fin de que desde este puedan iniciarse patrones.

Una vez que el usuario está conforme con sus bases de conocimiento y ha explorado patrones/pathways/caminos, entonces es posible aprovechar las facilidades que ofrece el menú general de análisis; disponible en la opción 2 del menú inicial (ver parte baja de la Tabla 1.4). La Tabla 1.8 muestra el resultado de acceder a tal opción. Tal menú ofrece varias opciones, entre ellas las que hacen posible las salidas presentadas en las Tablas 1.5 y 1.6, correspondientes a las opciones 4 (búsqueda de patrones para ligandos indicados por el usuario) y 8 (búsqueda de cadenas de patrones conectados).

La Tabla 1.4 muestra que el sistema provee un primer grupo de opciones de análisis. Estas opciones incluyen la número 2, diseñada para el análisis general de los RRG. La Tabla 1.8 muestra el resultado de acceder a tal opción. Las opciones 4 y 8 visibles en la Tabla 8, conducen a resultados del tipo ilustrado en las Tablas 1.5 y 1.6, respectivamente. En general, las opciones presentadas en la Tabla 8 proporcionan los siguientes modos de consulta: 1) dado un ligando, qué receptores lo reconocen; 2) identificar complejos en la red y proponer sus posibles roles regulatorios; 3) dado un receptor, a qué complejos se une; 4) dado un ligando, a cuáles patrones conduce y de qué tipo regulatorio son; 5) Dado un receptor, a qué motivos de ADN se une y a qué patrones conduce; 6) dado un receptor, qué ligandos lo reconocen; 7) dado un receptor, cuál de sus ligandos podría considerarse agonista, antagonista o de función mixta; 8) dado un objeto, encontrar grupos de patrones conectados del tipo estimulante / inhibidor, que podrían estar asociadas con él; y 9) dado un objeto, mostrar sus nombres, sinónimos, árboles MESH y GeneOntology (representaciones textuales, no gráficas).

Tabla 1.6. Ejemplo de una posible subred estimulatoria/inhibitoria para el SARS-CoV-2.

Pathway=>
'CD4',bind,'importin';'importin',associate,'CD8';'CD8',associate,'JAK3';'JAK3',associate,'MHC';'MHC',associate,'STAT';'STAT',regulate,'SARS-CoV-2'

Linking events: 'SARS-CoV-2',affect,'MHC'; 'SARS-CoV-2',mediate,'MHC'; 'SARS-CoV-2',change,'MHC'; 'SARS-CoV-2',bind,'MHC'; 'SARS-CoV-2',recognize,'MHC';

Pathway=> 'MHC',bind,'STAT';'STAT',bind,'STAT'

Linking events: 'STAT',express,'ORF6';

Pathway=> 'ORF6',bind,'STAT';'STAT',inhibit,'SARS-CoV-2'

Nota: Este ejemplo sugiere que el FT STAT1 podría estar involucrado tanto en la estimulación como en la inhibición de la replicación del SARS-CoV-2. Los posibles eventos de enlace sugieren cómo podrían conectarse ambos escenarios.

Tabla 1.7. Opciones disponibles para la re-ejecución de las etapas de modelado.

1.- Generar combinaciones de palabras clave
2.- Búsqueda de PubMed IDs
3.- Descargar abstracts desde PubMed
4.- Vaciado de ontologías a formato prolog (.pl)
5.- Generar resúmenes
6.- Generar base de conocimiento
7.- Proponer identidad de los objetos
0.- Volver

Tabla 1.8. Opciones disponibles para el análisis de la información de una RRG.

1.- Búsqueda de receptores.
2.- Búsqueda de complejos.
3.- Búsqueda de proteínas vinculadas a receptores.
4.- Búsqueda ligando/receptor/caminos.
5.- Búsqueda de motivos.
6.- Búsqueda de ligandos.
7.- Clasificar tipo de ligando.
8.- Buscar cadenas de pathways conectados.
9.- Consultar árboles de un objeto.
0.- Volver

A continuación un ejemplo de ejecución del sistema, acompañado de una descripción detallada de las distintas bases de conocimiento que ello genera.

2. Biopatrones. Ejecutando un experimento.

Esta sección describe cómo acceder a biopatrones y ejecutar un experimento. También se describen brevemente sus salidas (bases de conocimiento) y las etapas en las que estas se generan. Biopatrones es de libre acceso y puede ser descargado desde: https://github.com/biopattersg/biopatternsg. Pueden verse las instrucciones para instalar el sistema en el Anexo B de este documento. Para efectos de este documento emplearemos el sistema desde un servidor, a fin de facilitar el rápido acceso y estudio del mismo. Por favor no dude en dirigirnos un correo si desea nuestro apoyo (josesmooth@gmail.com o jacinto.davila@gmail.com). Se ha dispuesto además una serie de demos que describen las características generales del sistema y sus modos de uso. Puede accederse a ellos en Biopatternsg: Demos - Playlist. Si UD posee modo de acceder al sistema le recomendamos hacerlo ahora. En todo caso esta sección le servirá para hacerse de una buena idea sobre el modo de trabajar con el sistema. El anexo B de este documento describe cómo instalar el sistema y cómo accederlo, tanto en entornos Linux como Windows. En el caso de entornos Windows se recomiendan los siguientes programas: 1) puTTY, para acceder al servidor en modo terminal y 2) WinSCP, para acceder al servidor en modo explorador de archivos. El modo terminal es el empleado en esta guía. El modo explorador le permitirá desde Windows el acceso a las diferentes carpetas en el servidor en modo gráfico. Se recomienda ver los videos en los que se describen como poner a correr un experimento, tanto desde Linux como desde Windows. A continuación explicamos cómo realizar precisamente eso.

Biopatrones requiere de algunos datos iniciales para correr un experimento de modelado en el contexto de una red (en este ejemplo COVID-19). El modo de proveer tales datos consiste en colocar en una carpeta llamada ***data,*** las distintas carpetas de datos que corresponden a cada una de las redes en estudio. La carpeta data está ubicada bajo la carpeta del sistema, llamada **biopatternsg**. Para este ejemplo, la carpeta ***data*** contiene varias redes, siendo COVID-19 una de ellas; bajo esta última se desglosan otras tres denominadas COVID-19, COVID-19-Fármacos y COVID-19-IMMUNOLOGY. El caso data/COVID-19/COVID-19 se refiere a un experimento en el que solo interesa modelar interacciones de objetos relacionados con el COVID-19, mientras que data/COVID-19/COVID-19-Fármacos se refiere a un experimento en el que se quieren ver posibles relaciones entre objetos vinculados al COVID-19 y fármacos específicos. Por último, el experimento data/COVID-19/COVID-19-IMMUNOLOGY, contiene nombres de objetos del COVID-19 interactuando con objetos propios del sistema inmunológico. Cada carpeta, digamos data/COVID-19/COVID-19-IMMUNOLOGY, contendrá los siguientes archivos (un ejemplo de la carpeta ***data*** se incluye en el material suplementario):

1. Región reguladora. Este archivo debe contener el trozo de ADN que se considere define a la región reguladora de la transcripción del objeto de interés (COVID19, en este caso). El archivo se llama en este ejemplo “COVID19RegProm”.
2. Objetos del experto. Listado de todos los objetos que se considere deben ser parte de la red a modelar. Se definen allí los identificadores uno debajo del otro. El archivo se llama “expert_objects.txt” (ver ejemplo).
3. Proteínas homólogas (**opcional**). Listado de identificadores de las proteínas homólogas del objeto de interés que se considere deben incluirse en la red. El archivo se llama “homologous”. Se definen allí los identificadores uno debajo del otro (ver ejemplo archivo anterior).
4. Literatura del experto (**opcional**). En este caso se listan los PubMed IDs de los abstracts del usuario que desea que sean incluidos en el modelado de la red, sumándose a los abstracts que el sistema colecta automáticamente. El archivo debe llamarse "pubmed_IDsExp" (ver ejemplo).

2.1 Traza de ejecución y bases de conocimiento generadas.

La Tabla 2.1 muestra el modo de iniciar experimentos en el sistema. Según muestra la Tabla 2.1, el sistema despliega el listado de redes para las cuales se han proporcionado datos, o ya se han desarrollado experimentos. Una vez seleccionada alguna red (digamos COVID-19), el sistema lista los distintos procesos/experimentos de modelado que puedan haberse desarrollado para ella, ofreciendo la oportunidad de 1) integrarlos o 2) seleccionar algún experimento en particular. Si se elige integrar (opción I), entonces todas las bases de conocimiento disponibles se integran, generando un solo modelo para la totalidad de la red. En caso de elegir algún proceso particular (digamos 3) entonces se despliegan los resultados correspondientes para tal experimento. En caso de que ningún modelado se haya desarrollado para el experimento seleccionado, el sistema solicita los parámetros necesarios a tal fin. La Tabla 2.1 corresponde a este último caso.

2.1.1 Ejecutando un experimento.

El primer paso en la ejecución de un experimento es el acceso al servidor. Si UD ha realizado una instalación local en su máquina Linux, entonces es buen momento para dirigirse a la carpeta donde se realizó la instalación. En caso de acceder a un servidor UD debe haber recibido por correo los detalles de acceso correspondientes.

Una vez se ha dado forma a los requerimientos de un experimento en la carpeta ***data***, el sistema es invocado accediendo a la carpeta *biopatternsg*. Desde allí debe ejecutarse el comando indicado más abajo. Se recomienda ejecutar primero el comando *screen*, útil para recuperar posteriormente el experimento en curso. Ello debe llevarnos a la vista indicada en la Tabla 2.1 que muestra cómo parametrizar un nuevo experimento. El orden de los procesos listados puede diferir, asegúrese de seleccionar la opción <COVID-19-IMMUNOLOGY>. La intención es que UD provea los datos que ilustra la Tabla 2.1.

```
$ screen
$ java -jar biopatternsg.jar
```

Una vez elegido el idioma de su agrado podrá ver una pantalla similar a la presentada en la Tabla 2.1. Puede verse en la Tabla 2.1 que para un experimento nuevo deben indicarse parámetros. Tales parámetros corresponden a:

1. Región promotora. Corresponde a la secuencia de ADN que el usuario considera que es la región promotora (o de regulación) del objeto biológico de interés.. En este caso se provee en el archivo COVID-19RegProm. (ver *mc/data/COVID-19/COVID-19-IMMUNOLOGY/COVID-19RegProm*). Observe el contenido de la carpeta ***data*** en el servidor.
2. Número máximo de complejos desde PDB. Corresponde a la cantidad de complejos proteínicos relacionados con algún objeto de la red, a ser descargados desde PDB. En este caso se ha elegido no emplear esta opción.
3. Número de niveles de búsqueda. Corresponde a la cantidad de niveles que se explorarán en la construcción del modelo de la red (ver Figura 1). En este caso se ha elegido no emplear esta opción.

4. Índice de confiabilidad en TFBind. Corresponde al umbral de confiabilidad con el que se seleccionarán los FTs detectados en la región de regulación. Este parámetro indica la probabilidad de anclaje sobre el ADN suministrado, según la cual se elegirán FTs para sumarlos a los objetos de la red. En este caso se elige agregar a la red en construcción solo FTs con probabilidad igual o mayor a 0.98.
5. Cantidad máxima de PubMed IDs para cada búsqueda. Corresponde a la cantidad de PubMed IDs que se tomarán para cada par de palabras clave consultadas. Para el caso ilustrado, si se busca la combinación de palabras JAK3 y STAT, entonces solo se tomarán los 100 primeros PubMed IDs, que PubMed provea, en el que ambos objetos sean mencionados.
6. Referencias PubMed ID del Experto. Permite indicar si se proveerá un archivo con PubMed IDs seleccionados por el usuario. Se indica el nombre del archivo que los contiene (ver *mc/data/COVID-19/COVID-19-IMMUNOLOGY/pubmed_IDsExp*).
7. Nombre cortos o largos. Se define si se emplearán identificadores de objetos versión corta o larga en la búsqueda de PubMed IDs. Es normal que los objetos tengan asociados acrónimos o mnemónicos; responder sí implica que sólo se emplearán estos últimos.

Tabla 2.1. Un resumen de las opciones iniciales de biopatrones.

```
==== Redes disponibles ====
1.- BAXS
2.- COVID-19

Seleccione una opción: 2

Red: COVID-19

==== Procesos disponibles ====
1.- COVID-19
2.- COVID-19-Fármacos
3.- COVID-19-IMMUNOLOGY
4.- COVID-19-IMMUNOLOGY-I

Seleccione una opción: 3

biopatternsg

--------------------
Nuevo proceso:
--------------------

Proporciones los parámetros necesarios:

*Nombre de archivo región promotora:  COVID-19RegProm
*Índice de confiabilidad en TFbind (0-100): 99
*Número máximo de objetos PDB :0
*Número de niveles de búsqueda:0
*¿Desea agregar PubMed ID de experto?  ..S/N: s
*Indique el nombre del archivo PubMed IDs: pubmed_IDsExp
*Cantidad máxima de PubMed Ids para cada búsqueda: 100
*Usar nombres cortos para generar combinaciones de palabras clave...  S/N: s

Guardando configuración… ok.

Buscando información complementaria para los objetos indicados por el experto… .....
```

Una vez haya proporcionado los datos el experimento iniciará y tomará cierto tiempo que este termine. El tiempo que un experimento requiere dependerá de la cantidad de objetos de sus experimentos y de los parámetros indicados; tiempo que puede variar desde unas horas hasta varios días. Si su experimento se detiene por alguna razón, el sistema es capaz de

recuperarse desde el punto estable más cercano al momento de la interrupción. Ejecute el siguiente comando para dejar su experimento corriendo en el servidor (sin soltar las teclas `<ctrl> + <a>`).

```
$ <ctrl> + <a> + <d>
```

Para recuperarlo ejecute:

```
$ screen -r
```

Vuelva ejecutar el comando `<ctrl> + <a> + <d>` para salir del experimento.

El comando *screen* le permite tener varios experimentos corriendo y recuperarlos en cualquier momento. Si UD tiene más de un experimento corriendo, el comando *`screen -r`* le devolverá los identificadores asociados a cada uno. Si UD ejecuta *`screen -r <identificador>`*, el comando le devolverá el experimento correspondiente.

Ejecute el sistema de nuevo, pero ahora para explorar un ejemplo de lo que obtendrá al finalizar el experimento que recién echó a andar.

```
$ screen
$ java -jar biopatternsg.jar
```

Seleccione desde la salida ilustrada en la Tabla 2.1, la opción <COVID-19-IMMUNOLOGY-I>. Note que este experimento solicita hasta 2000 abstracts por cada par de objetos consultados a pubmed; lo que implica que requirió entre 4 a 7 días para finalizar. En el caso de su experimento este debería tardar no más de un día.

Digamos que UD quiere reproducir los resultados mostrados en las Tablas 1.3, 1.5 y 1.6, empleando el experimento <COVID-19-IMMUNOLOGY-I>. El procedimiento sería el siguiente:

Para la Tabla 1.3, seleccione la opción 3 del menú inferior en la Tabla 1.4. Tal opción corresponde a la inferencia de patrones. Provea los parámetros que se indican en la Tabla 2.2. Una vez haya finalizado la búsqueda de patrones el sistema mostrará un resumen de los parámetros del experimento y la cantidad de patrones hallados (en este caso 401). El resultado de la búsqueda genera en este caso tres archivos: 1) pathways.txt, que contiene los 401 patrones hallados; 2) *eventsDoc.txt*, que contiene los distintos eventos que dan forma a los patrones, acompañados a cada uno con las distintas oraciones que desde los abstracts les dan soporte conceptual, y 3) *pathwaysDoc.txt*, que lista los patrones acompañados de los eventos y oraciones que dan forma a cada patrón. Los archivos mencionados son colocados por el sistema en la ruta *minery/networks/COVID-19/COVID-19-IMMUNOLOGY-I*. Los archivos también pueden ser visualizados en *mc/minery/networks/COVID-19/COVID-19-IMMUNOLOGY*. La Tabla 1.3 corresponde a uno de los 401 patrones descritos en *pathwaysDoc.txt.* Se recomienda copiar el patrón y buscarlo en tal archivo. La Tabla 1.3 es una versión simplificada del

patrón que UD hallará al hacer la búsqueda. Algunos eventos tienen su documentación en otro patrón, por lo que se indica el número de patrón al cual ir para leer las oraciones que lo soportan.

Tabla 2.2. Ejecución de la búsqueda de patrones/pathways.

```
BioPatternsg
minery/networks/COVID-19/COVID-19-IMMUNOLOGY-I

Inferencia de patrones
*¿Desea restringir los pathway a un listado de objetos específicos? ..S/N: s
*Ingrese un listado de objetos separados por (,): Ejemplo: EGF,EGFR,Ras,CREB,SST
JAK3,STAT,MHC,SARS-CoV-2,ORF6,importin,CD4,CD8
*Desea agregar objeto de cierre a los pathway?  ..S/N:s
*Ingrese el nombre del objeto:STAT
*¿Desea agregar otro objeto de cierre?  ..S/N:s
*Ingrese el nombre del objeto:SARS-CoV-2
*¿Desea agregar otro objeto de cierre?  ..S/N:n
*¿Desea usar una base de conocimiento reducida? ..S/N:s

BioPatternsg
minery/networks/COVID-19/COVID-19-IMMUNOLOGY-I
Inferencia de patrones.....

*Región promotora:              COVID-19RegProm
*Cantidad de complejos:         0
*Número de niveles:             0
*Confiabilidad TFBind:          99
*Cantidad máxima de pubmed IDs:  2000
*Archivo de objetos del Experto:
data/COVID-19/COVID-19-IMMUNOLOGY/pubmed_IDsExp

Resultados:

Objetos minados:           39
Combinaciones realizadas:  10521
Pubmed Id encontrados:     76881
Eventos encontrados:       3908
Patrones encontrados:      401

Seleccione una opción:
1.- Crear un nuevo proceso.
2.- Ir al menú análisis de RRG.
3.- Inferir patrones.
4.- Actualizar la base de conocimiento de eventos.
5.- Documentar patrones de regulación.
6.- Reanudar desde.
0.- Volver
```

En el caso de Tabla 1.5, elija la opción 3 del menú inferior en la Tabla 2.2 para acceder al menú general para el análisis de la red en estudio. Una vez allí seleccione la opción 4, correspondiente a la búsqueda de patrones para un ligando y un receptor datos. Suministre ORF6 como ligando y STAT como receptor. En este ejemplo hemos convenido para efectos de exploración que el FT STAT cumpla con el rol de ser el receptor con el que ORF6 podría empezar patrones de regulación. Esta opción muestra sus resultados solo en pantalla, próximamente sus resultados serán exportados a un archivo llamado ligando-receptor-patrones.txt.

En el caso de la Tabla 1.6, regrese al menú mostrado en la parte inferior de la Tabla 1.8 y elija la opción 8, correspondiente a la búsqueda de patrones conectados. Desde la Tabla 1.6, copie el patrón allí mostrado y suminstrelo como patrón inicial. Este patrón en principio fue elegido desde el archivo *pathways.txt*, sugiriendo que el usuario halló en él algo de interés; lo que le motiva para usarlo como patrón inicial en la búsqueda de subredes.

Una vez indicado el patrón inicial, el sistema le preguntará hasta cuantos objetos permitirá UD que estén presentes en los patrones que se conectarán al patrón inicial. Suele ser un número entre 3 a 8, pero puede ser cualquier número; se recomienda aumentar gradualmente el número hasta obtener subredes. La salida de esta opción se guarda en un archivo llamado chainsPathways.txt. Recomendamos que lo renombre para que no lo pierda en próximas ejecuciones de esta opción. Aquí hemos optado por ponerle el nombre del objeto que cierra los patrones; en este caso SARS-CoV-2 (ver el archivo SARS-CoV-2_chainsPathways.txt, material complementario). Note que se trata de patrones que aparentemente primero estimulan la replicación del virus y luego la inhiben.

2.1.2 Ajustando la información para obtener resultados.

Antes de que puedan generarse resultados como los vistos anteriormente, deben hacerse algunos ajustes a la información que el sistema organiza automáticamente. Los dos ajustes principales corresponden a los archivos *minedObjects.txt* y *pathwaysObjects.pl*.

El archivo *minedObjects.txt* contiene los nombres y sinónimos separados por ';', asociados a los objetos de interés que se listan en la Tabla 1.2 (ver ejemplo del archivo en *mc/minery/networks/COVID19/COVID19-IMMUNOLOGY)*. El ajuste que corresponde a este archivo consiste en asegurarse de que cada objeto sólo tenga asociada una línea descriptiva. Tal línea debe incluir en primer lugar el nombre principal del objeto estandarizado por la comunidad seguido de los sinónimos que se hayan podido organizar automáticamente para él. Note que la Tabla 1.2 incluye nombres como JAK3 y JAK. Esto suele realizarse cuando se cuenta con dos nombres para un mismo objeto, que resulten igual de importantes para el usuario. Al colectar la información desde la internet, esto generará dos líneas descriptivas en *minedObjects.txt*. Es tarea del usuario asegurarse de que ambas líneas queden organizadas en una sola, a fin de que la base de conocimiento no posea eventos redundantes. Si UD indica no más de un identificador por objeto en el archivo expert_objects.txt, entonces este ajuste no es necesario. Otra situación para ajustar *minedObjects.txt* corresponde al caso en que el sistema haya fallado en hallar sinónimos para algunos de los objetos. En tal caso deberá incluirlos UD indicando primero el nombre principal, seguido de posibles sinónimos separados por ';' sin saltos de línea. Otra razón por la que debe revisarse el archivo *minedObjects.txt* es para asegurarse de que ningún sinónimo aparezca más de una vez. Si es el caso, corresponde al usuario elegir donde dejarlo. En caso de que el sistema falle hallando sinónimos para un identificador, este repetirá el nombre del objeto como único sinónimo (permita esa excepción). Cualquier cambio que el usuario realice en *minedObjects.txt,* implica que debe generarse de nuevo la base de conocimiento de eventos *kBase.pl*. Esta tarea debe realizarla el usuario a través de la opción 6 del menú descrito en la parte baja de la Tabla 2.2.

En lo que corresponde al archivo *pathwaysObjects.pl*, el usuario debe verificar si los roles asignados a los objetos son los correctos (ver el archivo en la ubicación indicada más arriba). Pueden verse en el archivo mencionado roles como ligando, proteína, factor de transcripción o receptor. Estos roles guían la búsqueda de patrones. Un patrón ha de empezar con un ligando enlazado a un receptor o mediante un receptor que se enlaza a otro receptor; siguiendo a este primer evento una serie de eventos intermedios que corresponden a proteínas conectándose entre sí. Los eventos intermedios terminan con un evento de cierre en el que un receptor o un FT, se enlaza a un objeto de interés. El usuario puede jugar con los roles,

definiendo así cuáles objetos pueden empezar caminos y cuáles pueden cerrarlos. Se espera agregar otros roles en la medida en que los usuarios sugieran nuevas restricciones para explorar los patrones.

Realizados los ajustes recién mencionados, el usuario queda en la posibilidad de explorar la inferencia de patrones y subredes, tal como se ha mostrado hasta ahora.

2.1.3 Depurando la base de conocimiento de eventos kBase.pl.

Una tarea que típicamente deberá ser considerada al analizar la base de conocimiento de eventos, los patrones y subredes, inherentes a un experimento, es la de depurar los eventos de regulación. Tal depuración debe realizarse manualmente, siendo su finalidad eliminar los falsos positivos que normalmente estarán presentes. Generalmente UD estará interesado en un subconjunto de objetos de la red, los patrones y las subredes que los conectan, por lo que bien puede enfocarse en los eventos correspondientes. En la Tabla 2.2 puede verse que el interés gira alrededor de los objetos JAK3, STAT, MHC, SARS-CoV-2, ORF6, importin, CD4, CD8 y que para ellos logran determinarse 401 patrones. Veamos cómo proceder si deseamos que los patrones no contengan falsos positivos.

Tabla 2.3. Algunos ejemplos sobre cómo etiquetar los eventos de regulación descritos en *eventsDoc.txt.*

event('MHC',bind,'STAT'):F
Both isoforms can downregulate MHC class II, however they differ in a number of other immunomodulatory properties, such as the ability to bind the IL10 receptor and induce signaling through STAT3.

event('MHC',recognize,'STAT'):F
Along with human leukocyte antigen gene encoding B,51 (HLA-B,51) and areas including the major histocompatibility complex class I, genome-wide association studies have recognized numerous other BD susceptibility genes including those encoding interleukin (IL)-10 , IL-12 receptor β 2 (IL-12RB2) , IL-23 receptor (IL-23R) , C-C chemokine receptor 1 gene, signal transducer and activator of transcription 4 (STAT4) , endoplasmic reticulum aminopeptidase (ERAP1) , and genes encoding killer cell lectin-like receptor family members (KLRC4-KLRK1).

event('MHC',recognize,'KLRC4'):U
Along with human leukocyte antigen gene encoding B,51 (HLA-B,51) and areas including the major histocompatibility complex class I, genome-wide association studies have recognized numerous other BD susceptibility genes including those encoding interleukin (IL)-10 , IL-12 receptor β 2 (IL-12RB2) , IL-23 receptor (IL-23R) , C-C chemokine receptor 1 gene, signal transducer and activator of transcription 4 (STAT4) , endoplasmic reticulum aminopeptidase (ERAP1) , and genes encoding killer cell lectin-like receptor family members (KLRC4-KLRK1).

event('MHC',detect,'STAT'):F
Statistical studies of associated alleles detected on each microsatellite locus showed that the pathogenic gene for Behçet disease is most likely found within a 46-kb segment between the MICA and HLA-B genes.

Nota: El etiquetado de este ejemplo no ha sido realizado por un biólogo.

Cada vez que se solicita al sistema que infiera patrones, este genera un archivo llamado *eventsDoc.txt*, que contiene la totalidad de los eventos que están presentes en la colección de patrones inferidos (ver archivo *eventsDoc.txt* en el material complementario). El archivo mencionado desglosa los eventos presentes en los patrones y las oraciones desde las que estos son modelados. Para depurar *eventsDoc.txt,* su tarea como usuario experto consiste en etiquetar cada evento como positivo (:P), falso (:F) o agregado por el usuario (:U). En el caso de los nuevos eventos aportados por el usuario, basta que este tome el evento que quiere corregir, lo copie, y luego proceda a modificar la estructura del evento de regulación, que acompaña a

la oración del caso. Luego de ello se agrega la etiqueta :U, para indicarle al sistema que tal evento ha sido modelado manualmente. El tercer evento de la Tabla 2.3 muestra un ejemplo; allí se muestra un evento modelado por el usuario a partir del segundo evento descrito. Note que el segundo evento es uno que el usuario etiquetó como falso al agregarle la etiqueta :F. Puede suceder que el usuario vea eventos nuevos en eventos que etiquetó como positivos. El procedimiento es el mismo en todo caso.

Una vez se hayan etiquetado todos los eventos en el archivo *eventsDoc.txt*, se solicita al sistema que actualice la base de conocimiento de eventos de regulación (archivo *kBase.pl*). Tal acción se realiza empleando la opción 4 del menú visible en la parte baja de la Tabla 2.2. Debe entonces procederse nuevamente a la inferencia de patrones (opción 3 del mismo menú). Es normal que al hacer estos cambios surjan patrones nuevos, por lo que es probable que el archivo *eventsDoc.txt* contenga nuevos eventos que requieran anotación manual. En tal caso el procedimiento deberá repetirse, hasta que el sistema le indique en el archivo *eventsDoc.txt,* que todos los eventos ya han sido debidamente etiquetados. Ello termina el proceso de depurado de *kBase.pl* para sus objetos de interés. Esto último es posible porque el sistema lleva un histórico de los eventos que UD ya ha etiquetado. La idea es que el usuario no deba repetir etiquetados que ya realizó. El histórico de eventos etiquetados queda almacenado en el archivo *eventsDoc-History.txt*.

Cuando se realiza la actualización de *kBase.pl*, y se verifica que *eventsDoc.txt* ya no contiene nuevos eventos para etiquetar, se está en la posibilidad de generar la documentación de los patrones de regulación, de tal modo que estos solo contengan eventos verdaderos positivos. Tal acción puede ejecutarse con la opción 5 del menú visible en la parte baja de la Tabla 2.2. La documentación de los patrones es guardada en *pathwaysDoc.txt*. Si UD no desea por los momentos etiquetar los eventos y aun así generar la documentación de los patrones, entonces solo debe etiquetar todos los eventos como positivos, guardar y proceder como ya se indicó. Tenga presente que en tal caso el histórico guardará falsos positivos. Por lo tanto, UD deberá eliminar el archivo *eventsDoc-History.txt,* cuando decida etiquetar los eventos; así el sistema empezará con UD desde cero. Recomendamos mantener respaldado el archivo kBase.pl. Si lo requiere, tal archivo puede generarse de nuevo desde la opción 6 del menú visible en la parte baja de la Tabla 2.2.

Volvamos al experimento que UD inició anteriormente. Para ello salga del experimento actual retrocediendo en los menús. Cierre el terminal en el que está corriendo el experimento actual, mediante el comando `<ctrl> + <d>`. Ahora, recupere con `screen -r` el experimento correspondiente a <COVID-19-IMMUNOLOGY>. Si por error se desconecta del servidor, vuelva a conectarse. Su experimento no se habrá perdido.

Su experimento quizás aún no ha terminado, por lo que debe estar en alguna de las fases correspondientes a la recolección de información. A continuación describimos una traza típica que le ayudará a ubicarse en cuál de las etapas se encuentra y el tipo de producto que esa etapa genera. Si al entrar al experimento lo consigue caído por alguna falla en el servidor, vuelva a ejecutar el experimento; el sistema lo reanudará en la etapa más cercana al momento de la falla. En lo que sigue, todos las etapas se describen apoyadas en el material complementario, tal como si UD estuviera observando su experimento una vez finalizado.

2.1.4. Productos generados en la exploración de conocimiento.

La Tabla 2.1 muestra que el proceso de búsqueda de información inicia justo después de indicar los parámetros necesarios; la Tabla 2.4 por otro lado, muestra una traza simplificada del proceso de exploración que sigue después; en este caso para el experimento COVID-19-IMMUNOLOGY de la red COVID-19. La Tabla 2.4 ilustra las diferentes etapas de modelado en la medida en que estas se van desarrollando. A continuación se comenta el contenido de esa tabla y se indican las salidas de cada etapa ejecutada. Las salidas del sistema obtenidas en este ejemplo están disponibles en el material complementario que acompaña a este documento. Tales salidas también están disponibles en el servidor en la ruta *minery/networks/COVID-19/COVID-19-IMMUNOLOGY-I/.*

1. Buscando información complementaria para los objetos del experto. Corresponde a la búsqueda de información complementaria para los objetos de la red proporcionados por el experto. Puede verse en *mc/minery/networks/COVID-19/COVID-19-IMMUNOLOGY/(minedObjects.pl y minedObjects.txt)*, que para cada objeto del experto se colecta información acerca de sus nombres actuales, posibles sinónimos, tejidos en los que es activo, entre otros detalles. Esto se realiza para todos los objetos de la red y corresponde a la búsqueda de información complementaria descrita en la Figura 2 del Anexo B.

2. Consultando FTs desde TFBind. Corresponde a la búsqueda de información complementaria referente a cada uno de los FTs que TFBind ha propuesto para la región reguladora del virus. Cada vez que se consulta información complementaria para algún objeto de la red, se definen su nombre estándar actual, sinónimos, ontologías MESH y GeneOntology. Los detalles relativos a las ontologías MESH y GeneOntology pueden verse en los archivos *ontologyMESH.pl* y *ontologyGO.pl*, disponibles en *mc/minery/networks/COVID-19/COVID-19-IMMUNOLOGY/*.

3. Consultando nuevos objetos PDB (no usado en este ejemplo). En este caso, para cada iteración en la construcción de la red, se consulta información complementaria para los nuevos objetos provenientes de PDB. No se ha usado esta opción en el presente ejemplo, por lo que la traza no muestra información al respecto.

4. Generando combinaciones de palabras clave. Una vez organizados los objetos de la red, se usan todos los nombres y sinónimos hallados para generar la colección de palabras claves, según las cuales se procederá a realizar la búsqueda de PubMed IDs.

5. Búsqueda de PubMed IDs. Se emplean las combinaciones de palabras clave previamente definidas para definir posibles PubMed IDs.

6. Descargando colección de abstracts. Definidos los PubMed IDs, se procede a la descarga de *abstracts* (ver *mc/minery/networks/COVID-19/COVID-19-IMMUNOLOGY/abstracts*). Puede verse en la carpeta indicada varios archivos en formato .zip (abstracts-xx.zip). Tales archivos contienen archivos de extensión html; cada uno de ellos guarda un número máximo de 700 abstracts. Estos archivos se procesan para generar los resúmenes (archivo *summaries.zip*), que luego se emplean para generar la base de conocimiento de eventos (kBase.pl) y la base de conocimiento documentada (kBaseDoc.txt).

Tabla 2.4. Traza de la ejecución del sistema (experimento COVID-19-IMMUNOLOGY)

```
Buscando información complementaria para los objetos indicados por el experto…
JAK
JAK3
.......

Consultando FTs desde TFBind….
   5 factores de transcripción encontrados
   Buscando información complementaria para los FTs .. ok

Generando combinaciones de palabras clave ... ok

Búsqueda de PubMed IDs ... ok

Descargando colección de abstracts... ok

Formateando ontologías, nombres y sinónimos de objetos ... ok

Generando resúmenes desde los abstracts.. ok

Generando base de conocimiento ... ok

Inferencia de patrones:

¿Quiere limitar los objetos en los patrones a una lista específica? ..S / N:  s

Proporcione una lista de símbolos de objetos separados por ",": por ejemplo: EGF, EGFR, Ras, CREB, SST
  JAK3,STAT,MHC,SARS-CoV-2,ORF6,importin,CD4,CD8

¿Quieres proporcionar un objeto de símbolo para terminar los caminos? ..S / N: s

Proporcione el nombre del objeto: STAT

¿Quiere proporcionar otro objeto para finalizar los caminos? .. S / N: s

Proporcione el nombre del objeto: SARS-CoV-2

¿Quiere proporcionar otro objeto para finalizar los caminos? .. S / N: n

¿Quiere utilizar una base de conocimientos reducida? .. S / N: s

Infiriendo patrones
…. Ok.

Resultados:

Objetos minados:           39
Combinaciones realizadas:   10521
Pubmed Id encontrados:      76881
Eventos encontrados:        3908
Patrones encontrados:       401

Seleccione una opción:
1.- Crear un nuevo proceso.
2.- Ir al menú análisis de RRG.
3.- Inferir patrones.
4.- Actualizar la base de conocimiento de eventos.
5.- Documentar patrones de regulación.
6.- Reanudar desde.
0.- Volver
```

7. Formateando ontologías, nombres y sinónimos de objetos. Toda la información colectada para todos los objetos, definida en *minedObjects.txt* y en las ontologias MESH y GeneOntology, se lleva a formato prolog. Ello genera los archivos: *minedObjects.pl, pathwaysObjects.pl, ontologiaGO.pl, ontologiaMESH.pl, well_know_rules.pl* (ver *mc/minery/networks/COVID-19/COVID-19-IMMUNOLOGY/*). Los archivos mencionados describen para cada objeto lo

correspondiente a: nombres y sinónimos; identidad (p.e., receptor, proteína, enzima, FT, ligando, etc); árboles de función molecular, de procesos biológicos y componente celular; definición y familias, según MESH; y, representación en forma de reglas, para apoyar el análisis automático de patrones y subredes.

8. Generando resúmenes desde los abstracts. Se genera un resumen general desde los abstracts descargados. Tal resumen contiene solo oraciones relacionadas con eventos de regulación (ver *mc/minery/networks/COVID-19/COVID-19-IMMUNOLOGY/*abstracts). Se genera un resumen para cada juego de abstracts.

9. Generando base de conocimiento. Desde los archivos de resúmenes se extraen los eventos de regulación y se construye la BC de eventos, que se emplea para inferir patrones (ver *mc/minery/networks/COVID-19/COVID-19-IMMUNOLOGY/*kBase.pl). El sistema genera además la base de conocimiento documentada *(kBaseDoc.txt),* que permite conocer las líneas de texto asociadas a cada evento en *kBase.pl*. El sistema también genera el archivo *kBaseR.pl*, útil para hacer más efectiva la exploración de eventos y patrones pues generaliza los eventos en *kBase.pl*. Para ello se usa la noción de relaciones sinónimas descritas en el archivo *relations-functions.txt*.

10. Inferencia de patrones. Permite restringir la búsqueda de patrones permitiendo que se indiquen los objetos que se desea sean parte de los mismos; también se indica si desea que se busquen objetos de cierre particulares en ellos. Esto último permite que se investiguen patrones de regulación para objetos específicos. También se puede indicar si se desea trabajar con una versión reducida de la BC de eventos; reducida en el sentido de que se eliminan eventos sinónimos, lo que disminuye el espacio de búsqueda.

11. Infiriendo patrones. Se infieren los patrones según los criterios anteriores (ver *mc/minery/networks/COVID-19/COVID-19-IMMUNOLOGY/pathways.txt)*.

12. Tan pronto la inferencia de patrones termina, se reporta información general sobre los procesos de consulta e inferencia desarrollados y se despliega el menú inicial, lo que permite realizar el análisis o la ejecución de nuevos procesos.

Hasta aquí lo relativo a la presentación del modo en que se puede interactuar con el sistema y el modo en que se puede realizar un experimento de modelado. Como se pudo observar, la opción 2 del menú visible en la parte baja de la Tabla 2.2 lleva al menú general de análisis que provee el sistema. De allí se ilustraron las opciones 4 y 8 (ver Tablas 1.5 y 1.6); otras opciones serán descritas en versiones venideras de este documento, pero al accederlas estas incluyen textos cortos que las describen. Le animamos a explorarlas y hacernos llegar sus preguntas o comentarios. También le sugerimos ver los demos descritos Biopatternsg : Demos - Playlist. A continuación dos secciones que le guiarán sobre cómo pudiera UD aprovechar mejor las potencialidades del sistema con una gestión del conocimiento más ajustada a lo que UD requiere.

2.2 El rol de las ontologías en biopatrones y cómo aprovecharlas.

En general, se ha seguido una estrategia de representación del conocimiento que procura facilitar su análisis automático; lo que esperamos facilite la incorporación de nuevas restricciones, acordes a nuevos requerimientos planteados por el usuario. Si el usuario propone una nueva restricción para navegar las bases de conocimiento, entonces tal estrategia debería ser relativamente fácil de incorporar a las opciones del sistema (eso esperamos). Considérese el siguiente ejemplo.

La Figura 2 muestra el resultado de una consulta MESH relativa a la hormona Somatostatin (SST). Allí pueden verse las seis ramas taxonómicas, que según MESH, corresponden a la citada proteína. Para aprovechar el conocimiento dispuesto en la Figura 2, hemos optado por una representación que favorece la interrelación de definiciones como las allí indicadas. biopatrones usa definiciones internas para representar lo descrito en la Figura 2, del siguiente estilo: *is_a('Somatostatin-28', 'Somatostatin'), is_a('Somatostatin', 'Pancreatic Hormones') e is_a('Pancreatic Hormones', 'Hormones')*. Tales definiciones (más otras de ese estilo), le permiten al sistema deducir que somatostatin es una familia de hormonas pancreáticas, de las cuáles somatostatin-28 resulta ser una de sus miembros. Esta aproximación facilita que el sistema pueda determinar automáticamente si somatostatin es una hormona o no; lo que puede conducir en un proceso de búsqueda a una solución en la que tal proteína participe. Nótese que para poder deducir que somatostatin pertenece a la familia de las proteínas, debe uno observar las diferentes ramas taxonómicas que MESH provee. En biopatrones, tal tarea es realizada empleando procesos automáticos de inferencia. La Figura 3 ilustra el modo en el que biopatrones representa internamente las taxonomías recibidas desde MESH.

Figura 2. Respuesta MESH a consulta inherente a Somatostatin.

En biopatrones, la representación interna del conocimiento descrito en la Figura 2, permite la definición de lo que hemos llamado el árbol de identidad para la proteína SST. Tal árbol puede verse en la Figura 3, elaborado manualmente, y corresponde a la representación gráfica del conocimiento taxonómico representado en biopatrones como un juego de relaciones del tipo *"is-a"*. Dado que el sistema hace esto disponible para todos los objetos de una red, entonces el usuario podría ver en ello oportunidades para indicar diversos modos de restringir, relacionar, o explorar los objetos en sus procesos de búsqueda.

Lo expuesto arriba para MESH también es cierto para GeneOntology. Puede verse en la Figura 4 parte del árbol GeneOntology para somatostatin. La Figura 4 sólo muestra seis de los posibles veinte nodos inmediatos a somatostatin (Figura 4, parte baja); la mayoría de ellos asociados a procesos biológicos. Invitamos al lector a imaginar la amplitud y complejidad del gráfico correspondiente. En biopatrones los árboles están disponibles lógicamente, no gráficamente; sin embargo, toda la potencialidad que estos gráficos sugieren, está al alcance del usuario para definir criterios (o restricciones) de búsqueda. Considérense todos los árboles para todos los objetos de una red. Si se definen criterios de interés para recorrer ese "bosque", entonces podrían explorarse posibles interrelaciones entre objetos de la red, que conduzcan a posibles soluciones; por ejemplo, tipos muy específicos de patrones o subredes, restringidos a tipos muy específicos de objetos. La dinámica de trabajo que proponemos aquí es que el usuario conozca estas potencialidades y, apoyado por los programadores, proponga sus requerimientos y restricciones, a fin de guiar exploraciones apoyadas en los árboles recién descritos.

2.3. Cómo podría el usuario escalar el experimento aquí descrito.

La Tabla 1.2 sugiere un modelado en el que participaron varios de los objetos que caracterizan a los virus SARS-COV y SARS-COV-2. Por otro lado, la Tabla 1.4 muestra que el índice de confiabilidad TFBind corresponde a 0,98, lo que implica una restringida participación de FTs en el experimento. También es posible ver que no se están usando los servicios de PDB y que además, tampoco se emplean homólogos de la secuencia de ADN suministrada. Los diversos parámetros mencionados pueden ser incorporados, según considere el usuario, y estos podrían ser empleados para ejecutar procesos de modelado independientes con orientaciones particulares. Considérese un escenario en el que se elige ejecutar tres procesos adicionales de modelado. El primer proceso podría estar enfocado en un listado ampliado de objetos SARS-COV y SARS-COV-2, que incluya las proteínas accesorias ORFx. En el segundo experimento, se podrían conservar los parámetros mostrados en la Tabla 1.4, pero optar por disminuir el índice de confiabilidad TFBind. El tercer proceso de modelado se podría enfocar en un listado detallado de proteasas, pero incluir un grupo muy específico de fármacos, que el usuario considere puedan inhibir su actividad. Una vez realizados los procesos de modelado, estos podrían integrarse para aprovechar las ventajas correspondientes. Cada experimento en sí mismo es exigente en cuanto a recursos de cómputo, por lo que resulta conveniente diseñar experimentos independientes que luego se integran. Considérese además que a la hora de analizar resultados puede uno realizar los experimentos por separado y, luego de haber "madurado" lo obtenido en cada caso, proceder a aprovechar las posibles ventajas de la integración.

Hemos expuesto en general las funcionalidades de un sistema que permite organizar bases de conocimiento, partiendo de un conjunto de identificadores de interés. Una vez que el sistema ha organizado el conocimiento, se hace posible acceder a un

conjunto de opciones que exploran posibles patrones y subredes de regulación, entre otras posibilidades. La dinámica aquí descrita es una en la que se procura la sistematización de procesos, pero que deja abiertas distintas posibilidades, para que el usuario participe activamente. Nuestra intención con este documento ha sido exponer a posibles interesados las potencialidades de nuestro sistema, con la finalidad de invitarlos a hacer equipo y explorar posibles escenarios de trabajo.

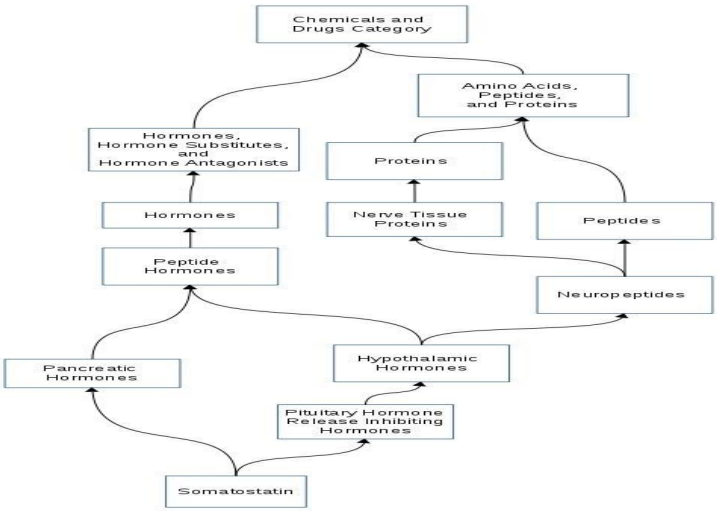

Figura 3. Respuesta MESH para Somatostatin desde biopatrones (manualmente elaborada desde las reglas internas).

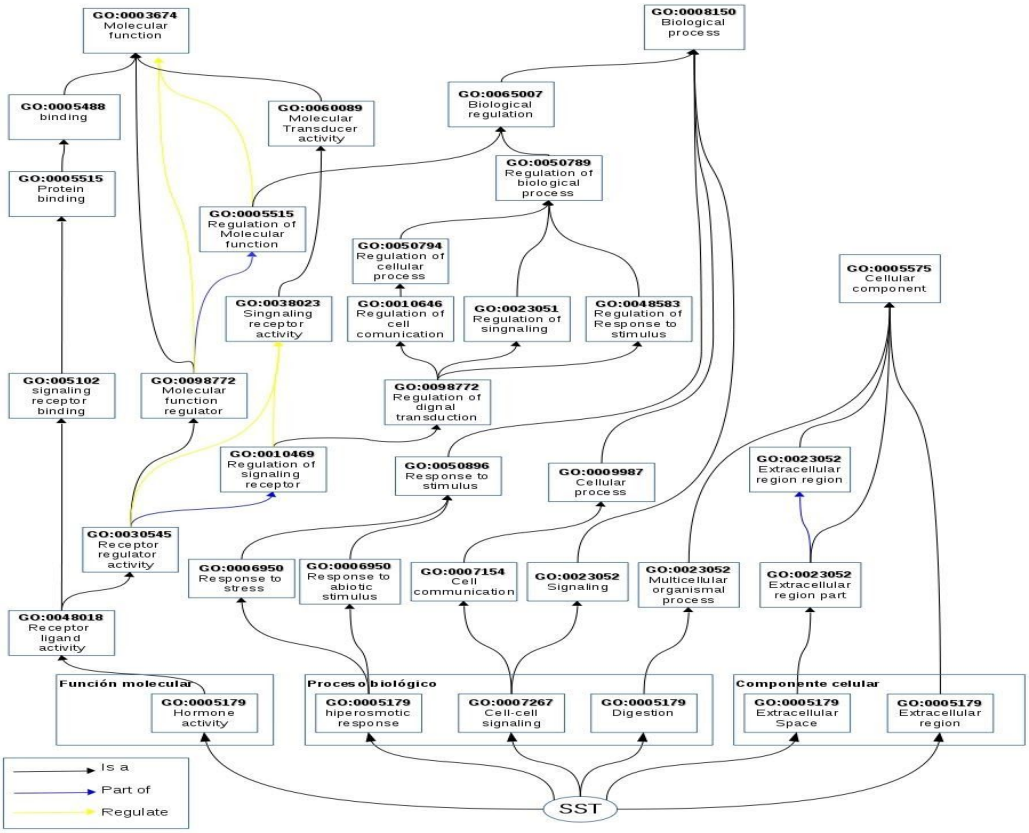

Figura 4. Vista parcial árbol GeneOntology para somatostatin desde biopatrones (elaborado manualmente).

Anexo A. Biopatrones. Sobre cómo se recolecta la información.

Las redes de regulación genética (RRGs) definen un área muy activa en el modelado informático de sistemas biológicos y están sugiriendo soluciones innovadoras a problemas muy diversos (Emmert-Streib et al, 2014). En tales redes, un producto particular (un transcrito), se relaciona con la regulación de otros objetos presentes en la misma red, o en redes diferentes. Por lo tanto, un evento de regulación puede activar un producto que a su vez participa en un evento, que activa o inhibe a otro producto. Es normal entonces que en tales redes la complejidad de las interrelaciones crezca muy rápidamente al pretender modelarlas. Para gestionar el conocimiento actual inherente a las RRGs, se han desarrollado estrategias informáticas destinadas a la descripción, organización, interrelación y análisis de los elementos que las constituyen. Entre tales estrategias se encuentran las ontologías (Demir E. et al, 2010; Aditya y Babita, 2017; Muñoz-Torres y Carbon, 2017) y los diagramas de procesos (Kitano et al, 2005; Kitano, 2015; Kitano, 2016); el primero orientado al análisis semántico de las redes y el segundo a la simulación de su dinámica molecular. Es de nuestro particular interés aquí, sistemas de modelado y análisis basados en la procesamiento inferencial de bases de conocimiento (Rougny et al, 2018). Elaborando sobre tales tendencias nos hemos propuesto la construcción automática de bases de conocimiento (BC), partiendo desde el cómo éstas se describen en la literatura científica. En concreto, hemos desarrollado un sistema que lee resúmenes científicos de documentos y produce representaciones de eventos de regulación, para una RRG de interés, que pueden ser analizados computacionalmente. Para ello nos hemos propuesto recolectar automáticamente la información que se pudiera requerir, acerca de los objetos presentes en una red y sus posibles modos de interacción.

Los objetivos del trabajo aquí presentado son: 1) la construcción automática de bases de conocimiento que describen redes de regulación genética (especies moleculares y sus posibles interacciones) y 2) el desarrollo de estrategias de análisis exploratorio, que pudieran guiar a nuevos hallazgos, o nuevas conclusiones, desde lo ya publicado. En cuanto a 1) el sistema desarrollado recopila información desde diversos repositorios y servicios disponibles en Internet, necesaria para responder consultas diversas, de las cuales hemos instrumentado algunos prototipos. En cuanto a 2), hemos diseñado un conjunto de representaciones para la información recolectada, que esperamos facilite el análisis automático y semiautomático de la misma. Nuestro propósito es responder a consultas del siguiente estilo: d*ado un ligando, qué tipo de interacciones proteína-proteína resultan cuando este ligando se une a un receptor conocido, y cuáles conducen a la activación (o inhibición) de la respuesta transcripcional de algún gen*. También consideramos preguntas como: *Para un par de proteínas dadas, ¿hay alguna subred en esta RRG, que describa patrones de regulación interconectados, en los que tales proteínas estimulen e inhiban su transcripción de alguna modo?*.

A los fines expuestos, hemos desarrollado la estrategia esquemática expresada en la Figura A.1. El modelado de una RRG inicia desde la región de regulación de la transcripción de algún objeto biológico, avanzando por niveles, hasta alcanzar objetos extra celulares (un medicamento, por ejemplo). Nuestra metodología inicia proponiendo factores de transcripción que podrían reconocer la región de regulación de una proteína u otro objeto biológico (un virus, por ejemplo). A partir de allí, el modelado se mueve hacia arriba integrando otros objetos, proporcionados por el usuario o consultados en Internet. La Figura A.1 muestra que otros objetos se pueden agregar a la red desde servicios como los proporcionados por el Protein Data Bank (PDB) (Rose et al, 2017) (Berman et al, 2014).

En este trabajo, la búsqueda de patrones de regulación se guía por definiciones como esta: *un patrón de regulación es aquel en el que un ligando reconoce un receptor, lo que desencadena una cascada de eventos regulatorios en los que pueden estar involucrados diferentes tipos de proteínas; cascada de eventos que se cierra mediante uno en el que un factor de transcripción reconoce un elemento de respuesta, y por lo tanto, activa o inhibe, la transcripción de un producto específico (ARN o proteína)*. En nuestro caso, el hallazgo de la solución consiste en determinar una colección de eventos regulatorios, que satisfagan las restricciones implícitas en la definición anterior. Habiendo organizado automáticamente y semi automáticamente, el conocimiento, según indican las Figuras A.1 y A.2, entonces los eventos que satisfacen restricciones específicas, se pueden explorar en una base de conocimiento como la que se muestra en la Tabla A.1. Nuestro objetivo es desarrollar un marco lógico y ontológico, en el que pueda ser posible recibir definiciones desde el usuario (como la anterior), programarlas y construir bases de conocimiento, lo suficientemente robustas como para ayudar en su resolución.

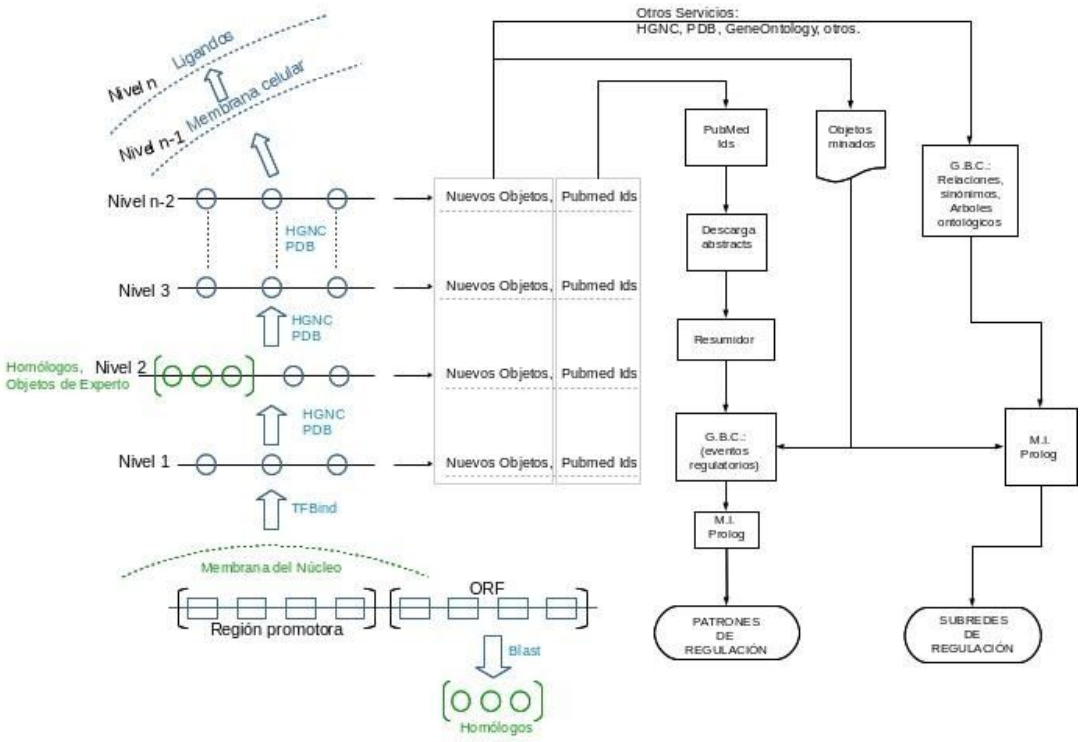

Figura A.1. Arquitectura del sistema y dinámica para el modelado semántico y el análisis de una RRG.

A.1. Una cadena de procesos

Para lograr los objetivos descritos, hemos implementado un conjunto de programas (o procesos) secuencialmente conectados (ver Figura A.2). La Figura A.2 muestra que cada nivel de exploración proporciona nuevos objetos acompañados de la información complementaria correspondiente. Una vez que se completan los niveles de modelado indicados por el usuario, se procede a la definición de posibles PubMed IDs relacionados con los nombres y sinónimos de los objetos de la red (Fiorini et al, 2017). Luego, se descargan los resúmenes relacionados con esos PubMed ID y se obtiene un resumen general, que contiene solo oraciones relacionadas con eventos regulatorios . Una vez que se define el resumen general, se genera automáticamente una BC de eventos del estilo descrito en la Tabla A.1. Posteriormente, la BC modelada se entrega a un sistema de inferencia, lo que hace posible el descubrimiento de posibles patrones y subredes. Tales patrones,

idealmente, deberían mostrar patrones que van desde la membrana celular (región cercana a los ligandos), hasta la membrana nuclear (región cercana a los factores de transcripción) (ver Figura A.1).

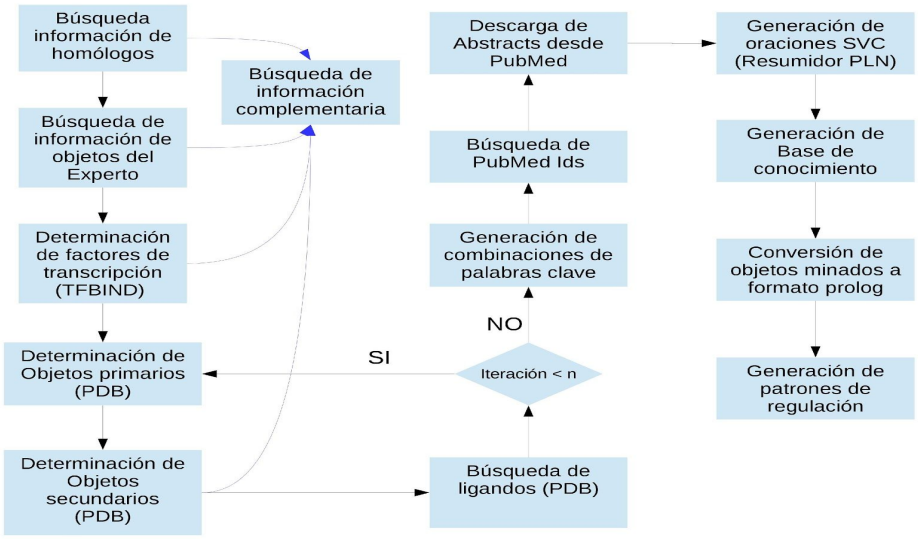

Figura 2. Secuencia de procesos propuesta para implementar el modelado semántico de una RRG.

Como se ha indicado, también interesa aquí determinar posibles subredes de objetos estrechamente vinculados. La Figura A.3 muestra un ejemplo tomado del sistema BAXS (Lu TT et al, 2000). Allí vemos dos proteínas principales, CYP7A1 y SHP. Para descubrir subredes como la que se muestra, nuestro enfoque se guía por la determinación de patrones de regulación que se conectan a través de eventos de enlace; lo que ofrece a quien explora la red, posibles alternativas de vinculación entre patrones particularmente interesantes. La idea principal es determinar las subredes en las que cada una de las proteínas de interés son estimuladas e inhibidas. Por ejemplo, si se encuentra una subred en la que CYP7A1 es estimulada (primero) e inhibida (después), y es posible observar en ella alguna otra proteína que medie entre ambos escenarios, entonces podrían determinarse otras subredes para tal proteína mediadora, que sugiera las condiciones que rigen su estimulación o inhibición. En el ejemplo, CYP7A1 se estimula e inhibe dependiendo de la presencia o ausencia de SHP, por lo tanto, es importante encontrar subredes en las que se estimule o inhiba SHP. Dos subredes coexisten en la Figura 2. Una que muestra las condiciones que estimulan e inhiben la presencia de CYP7A1, y otra que describe lo mismo para SHP. Este trabajo ofrece una opción de escaneo semiautomático para descubrir escenarios como el descrito.

A.2. Sobre el conocimiento recolectado

Las Figuras A.1 y A.2 ilustran varios procesos que guían la elaboración de diversas bases de conocimiento, necesarias para describir los diferentes objetos colectados en el proceso de modelado de una RRG. Tomemos la Tabla A.1 como ejemplo, correspondiente a un conjunto de eventos como el ilustrado en la Figura 3 (Lu TT et al, 2000). La Tabla A.1 muestra eventos en los cuales un sujeto afecta de alguna manera a un objeto y la sintaxis para representar esto es: *evento (sujeto, relación, objeto)*. Los sujetos y objetos visibles en la Tabla A.1, ilustran las especies moleculares colectadas en un proceso de modelado; considerando cada una de ellas, nuestro sistema construye cuatro BC adicionales, a fin de definir: 1) nombre, sinónimos y otras definiciones básicas, desde PubMed (Fiorini et al, 2017), Protein Data Bank (PDB) (Rose et al,

2017) (Berman et al, 2014), HGNC (HUGO Gene Nomenclature Committee) (Gray et al, 2016), GeneOntology (Thomas, 2017), Uniprot (Pundir, 2017), Mesh (Baumann, 2016) y Pathway Commons (Rodchenkov et al, 2019); 2) ontología MESH, 3) ontología GeneOntology (función molecular, procesos biológicos y componente celular); y 4) hechos lógicos, que establecen que un objeto satisface las restricciones que guían el análisis de patrones y subredes. A continuación un ejemplo que ilustra cómo se procede en general para modelar y analizar una RRG.

Tabla A.1. Un subconjunto de eventos regulatorios en la RRG del Sistema BAXS.

```
base([
event('cholesterol',regulate,'oxysterols'),
event('oxysterols',bind,'LXRa'),
event('oxysterols',activate,'LXRa'),
event('LXRa',associate,'RXR'),
event('LXRa',bind,'LXRE'),
event('LXRa',activate,'CYP7A1'),
event('CYP7A1',increase,'ba'),
event('ba',bind,'FXR'),
event('ba',activate,'FXR'),
event('FXR',associate,'RXR'),
event('FXR',bind,'FXRE'),
event('FXR',regulate,'SHP'),
event('SHP',associate,'LRH'),
event('LRH',bind,'LRHRE'),
event('LRH',inhibit,'SHP'),
event('LRH',inhibit,'CYP7A1')
]).
```

Nota: Esta BC ha sido modelada manualmente desde la literatura y corresponde a lo descrito en la Figura 3.

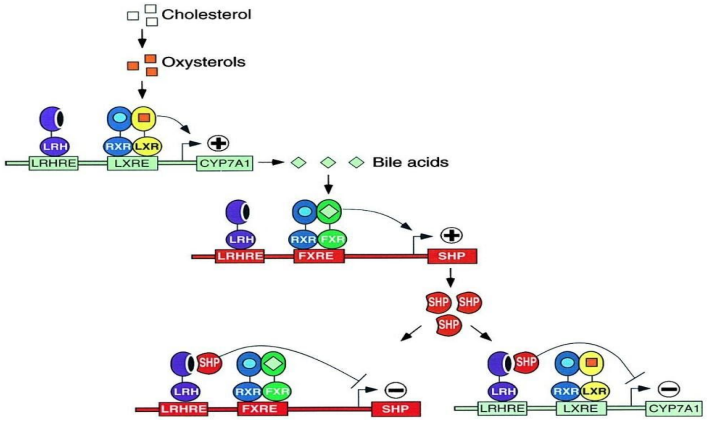

Figura A.3. Una subred regulatoria del sistema BAXS. Crédito: Lu TT, et al (2000).

Anexo B. Biopatrones: Instalación y ejecución.

Biopatrones corre en entornos Linux, pero una vez instalado en un servidor, también puede ser accedido desde entornos Windows. La instalación que aquí se describe se realizó en una máquina corriendo Debian 10. Se requieren para la instalación de biopatrones, Java 8 or superior, y Prolog versión 7.2.3 or superior. Aquí se describe la instalación de prolog usando apt, lo que implica a la fecha, la versión de 8.0.2. En cuanto a Java se ha usado OpenJDK 11, que es la versión que por omisión incluye Debian 10. En caso de ser usuario Windows, se recomiendan los siguientes programas para acceder a un servidor: puTTY, para acceder al servidor en modo terminal y 2) WinSCP, para acceder al servidor en modo explorador de archivos. El modo terminal es el empleado en esta guía. El modo explorador le permitirá desde Windows el acceso a las diferentes carpetas en el servidor en modo gráfico.

1. Instalación de Java.

   Acceda a un terminal como root o use el comando sudo, y ejecute los siguientes comandos:

   $ sudo apt update
   $ sudo apt install default-jdk

   Una vez finalizada la instalación, verifique su versión de java:

   $ java -version

   salida:
   openjdk version "11.0.9" 2020-10-20
   OpenJDK Runtime Environment (build 11.0.9+11-post-Debian-1deb10u1)
   OpenJDK 64-Bit Server VM (build 11.0.9+11-post-Debian-1deb10u1, mixed mode, sharing)

2. Instalación de prolog y JPL.

   Ejecutar los siguientes comandos:

   $ sudo apt-get update
   $ sudo apt-get install swi-prolog
   $ sudo apt-get install swi-prolog-java

   Editar ./.bashrc incluyendo las siguientes líneas, asegurándose de que cada export ocupe solo una línea:

   export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:
   /usr/lib/jvm/java-11-openjdk-amd64/lib/:
   /usr/lib/jvm/java-11-openjdk-amd64/lib/server/:
   /usr/lib/swi-prolog/lib/:
   /usr/lib/swi-prolog/lib/x86_64-linux/
   export LD_PRELOAD=/usr/lib/swi-prolog/lib/x86_64-linux/libjpl.so

3. Instalación de biopatrones (opción git). Dado que los procesos de modelado suelen requerir tiempo y recurso computacional, se recomienda su ejecución en un servidor. El sistema puede ser ejecutado en un computador personal pero este debe ser uno bien dotado de recursos. Se recomiendan máquinas con al menos 8 GB de RAM y procesadores de reciente generación.

   Cree un directorio llamado biopatternsg:

   ```
   $ mkdir biopatternsg
   ```

   Cambie su ubicación a ese directorio:

   ```
   $ cd biopatternsg
   ```

Ejecute los siguientes comandos:

```
$ git init

$ git remote add origin https://github.com/biopatternsg/biopatternsg.git

$ git fetch origin

$ git checkout runnable
```

Salida:

```
Branch 'runnable' set up to track remote branch 'runnable' from 'origin'.
Switched to a new branch 'runnable'
```

La salida anterior indica que ya se está sincronizado con la versión ejecutable del sistema.

Liste el contenido de la carpeta:

```
$ ls
```

Esta debe contener algo como:

```
biopatternsg.jar   % El sistema.
data         % Carpeta que contiene la data requerida para los diferentes
experimentos relativos a cada red.
language     % Textos empleados en los menús del sistema para cada idioma
lib          % Librerías de apoyo del sistema
minery       % Contendrá los modelos de cada red y la integración que de estos se
haya realizado.
LEEME        % Incluye estas instrucciones.
bioinformant_summarizer   % Sistema resumidor que permite el procesamiento de
abstracts.
scripts      % scripts varios para el desarrollo de consultas y búsquedas de
patrones de regulación
biopatrones-guia-del-investigador.pdf  % Guía actualizada sobre el uso del sistema.
biopatternsg-reseacher-guide.pdf % The user's system guide (updated version).
```

Ejecute el siguiente comando y elija el idioma de su gusto:

```
$ java -jar biopatternsg
```

Por favor, proceda según se describe en las secciones 1 y 2 de esta guía.

4. Instalación de biopatrones (opcion .zip). Para instalar biopatrones en esta opción, solo se requiere descargarlo, descomprimirlo y ejecutarlo.

   Descargue el archivo biopatternsg-runnable.zip desde el siguiente enlace, accediendo al icono CODE:

   Descomprima el sistema mediante el siguiente comando:

   $ unzip biopatternsg-runnable.zip

   Diríjase a la carpeta biopatternsg-runnable.

   La carpeta mencionada debe contener los siguientes elementos:

```
biopatternsg.jar   % El sistema.
data         % Carpeta que contiene la data requerida para los diferentes experimentos relativos a cada red.
language     % Textos empleados en los menús del sistema para cada idioma
lib          % Librerías de apoyo del sistema
minery       % Contendrá los modelos de cada red y la integración que de estos se haya realizado.
LEEME        % Incluye estas instrucciones.
bioinformant_summarizer   % Sistema resumidor que permite el procesamiento de abstracts.
scripts      % scripts varios para el desarrollo de consultas y búsquedas de patrones de regulación
biopatrones-guia-del-investigador.pdf  % Guía actualizada sobre el uso del sistema.
biopatternsg-reseacher-guide.pdf % The user's system guide (updated version).
```

Ejecute el siguiente comando y elija el idioma de su gusto:

```
$ java -jar biopatternsg
```

Por favor, proceda según se describe en las secciones 1 y 2 de esta guía.

Referencias

Aditya Khamparia, Babita Pandey (2017). Comprehensive analysis of semantic web reasoners and tools: a survey. Education and Information Technologies. November 2017, Volume 22, Issue 6, pp 3121–3145.

Baumann N (2016). How to use the medical subject hea dings (MeSH). Int J Clin Pract. 2016 Feb;70(2):171-4. doi: 10.1111/ijcp.12767. Epub 2016 Jan 13.

Berman, H.M., Kleywegt, G.J., Nakamura, H., and Markley, J.L. (2014). The Protein Data Bank archive as an open data resource. Journal of Computer-Aided Molecular Design. October, Volume 28, Issue 10, pp 1009–1014.

Bhalla U (2003). Understanding complex signaling networks through models and metaphors. Prog Biophys Mol Biol. 2003 Jan;81(1):45-65.

Demir E., Cary MP, Paley S., Fukuda K., Lemer C., Vastrik I., Wu G., D'Eustachio P., Schaefer C., Luciano J., Schacherer F., Martinez-Flores I., ..., Sander C., and Bader G.D., (2010). The BioPAX community standard for pathway data sharing. Nat Biotechnol. 2010 Sep;28(9):935-42. doi: 10.1038/nbt.1666. Epub 2010 Sep 9.

Emmert-Streib F., Dehmer M., and Haibe-Kains Benjamin (2014). Gene regulatory networks and their applications: understanding biological and medical problems in terms of networks. Front Cell Dev Biol. 2014 Aug 19;2:38. DOI: 10.3389/fcell.2014.00038.

Fiorini, N., Lipman, D. J., & Lu, Z. (2017). Towards PubMed 2.0. eLife, 6, e28801. doi:10.7554/eLife.28801.

Gevaert O, Van Vooren S, De Moor B (2007). A framework for elucidating regulatory networks based on prior information and expression data. Ann N Y Acad Sci. 2007 Dec;1115:240-8. Epub 2007 Oct 9.

Gray, K.A., Seal, R.L., Tweedie, S., Wright, M.W., and Bruford, E.A. (2016). A review of the new HGNC gene family resource. Human Genomics. Feb 3;10(1):6. DOI: 10.1186/s40246-016-0062-6. PMID:26842383.

Kitano H., Funahashi A., Matsuoka Y., and Oda K. (2005). Using process diagrams for the graphical representation of biological networks. Nat Biotechnol. 2005 Aug;23(8):961-6. DOI: 10.1038/nbt1111.

Kitano, H. (2015). Accelerating systems biology research and its real world deployment. NPJ Syst Biol Appl. 2015 Sep 28;1:15009. DOI: 10.1038/npjsba.2015.9.

Kitano, H. (2016). Artificial Intelligence to Win the Nobel Prize and Beyond: Creating the Engine for Scientific Discovery. AI Magazine, 37(1), 39-49. https://doi.org/10.1609/aimag.v37i1.2642.

Lu TT, Makishima M, Repa JJ, Schoonjans K, Kerr TA, Auwerx J, Mangelsdorf DJ (2000). Molecular basis for feedback regulation of bile acid synthesis by nuclear receptors. Mol Cell. 2000 Sep;6(3):507-15. DOI: 10.1016/s1097-2765(00)00050-2.

Munoz-Torres M, Carbon S (2017). Get GO! Retrieving GO Data Using AmiGO, QuickGO, API, Files, and Tools. Methods Mol Biol. 2017;1446:149-160. PMID: 27812941.

Pundir, S., Martin, M. J., & O'Donovan, C. (2017). UniProt Protein Knowledgebase. Methods in molecular biology (Clifton, N.J.), 1558, 41–55. doi:10.1007/978-1-4939-6783-4_2.

Rodchenkov I, Babur O, Luna A, Aksoy BA, Wong JV, Fong D, Franz M, Siper MC, Cheung M, Wrana M, Mistry H, Mosier L, Dlin J, Wen Q, O'Callaghan C, Li W, Elder G, Smith PT, Dallago C, Cerami E, Gross B, Dogrusoz U, Demir E, Bader GD, Sander C (2019). Pathway Commons 2019 Update: integration, analysis and exploration of pathway data. Nucleic Acids Res. 2019 Oct 24. pii: gkz946. doi: 10.1093/nar/gkz946. PMID: 31647099.

Rose P.W., Prlić A., Altunkaya A., Bi C., Bradley A.R., Christie C.H., Costanzo L.D., Duarte J.M., Dutta S., Feng Z., Green R.K., Goodsell D.S., Hudson B., Kalro T., Lowe R., Peisach E., Randle C., Rose A.S., Shao C., Tao Y.P., Valasatava Y., Voigt M., Westbrook J.D., Woo J., Yang H., Young J.Y., Zardecki C., Berman H.M., and Burley S.K., (2017). The RCSB protein data bank: an integrative view of protein, gene and 3D structural information. Nucleic Acids Res. 2017 Jan 4;45(D1):D271-D281. DOI: 10.1093/nar/gkw1000. Epub 2016 Oct 27.

Rougny, A., Gloaguen, P., Langonné, N. et al (2018). A logic-based method to build signaling networks and propose experimental plans. Sci Rep 8, 7830 (2018) doi:10.1038/s41598-018-26006-2.

T.Tsunoda, and T.Takagi (1999).Estimating Transcription Factor Bindability on DNA. BIOINFORMATICS, Vol.15, No.7/8, pp.622-630, 1999.

Thomas P. D. (2017). The Gene Ontology and the Meaning of Biological Function. Methods in molecular biology (Clifton, N.J.), 1446, 15–24. doi:10.1007/978-1-4939-3743-1_2.