Endo-Mining: herramienta web para la búsqueda automatizada de genes potencialmente relacionados con la endometriosis a través de minería de textos en PubMed

Jorge Vallejo Ortega

28/03/2021

Índice

1 Búsqueda en PubMed

PubMed es un recurso en línea de acceso público y gratuito consistente en una base de datos - en continuo crecimiento - que incluye más de 32 millones de citas y abstracts de literatura biomédica, tanto artículos (*MEDLINE* y *PubMed Central*) como libros (*Bookshelf*). Esta base de datos - en línea desde 1996 - fue desarrollada y sigue siendo mantenida por el Centro Nacional para la Información Biotecnológica (*National Center for Biotechnology Information*, NCBI), que forma parte de la Biblioteca Nacional de Medicina de los E.E.U.U. (*U.S. National Library of Medicine*, NLM) de los Institutos Nacionales de Salud (*National Institutes of Health*, NIH). Esta base de datos está especializada en publicaciones centradas en campos científicos relacionados con la salud y la biomedicina (National Library of Medicine (2021a), National Library of Medicine (2021b)).

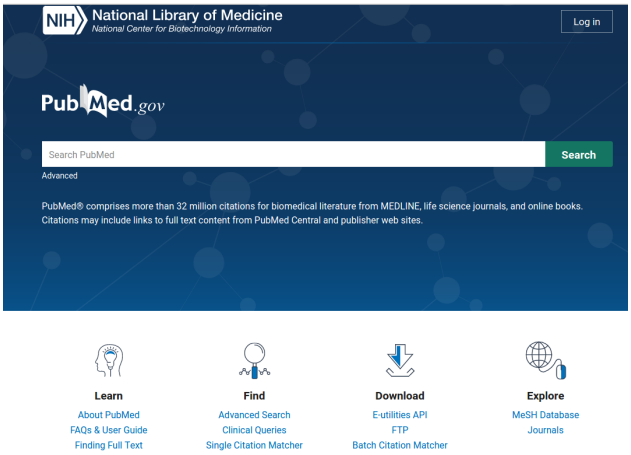


Figura 1: Página principal de PubMed. La barra de búsqueda destaca en medio de la imagen, indicando la finalidad principal de esta página.

Para realizar búsquedas es posible utilizar una serie de etiquetas con las que especificar si el término que hemos escrito debe buscarse como parte del título (etiqueta [TI]), el autor ([AU]), la revista ([TA]) o cualquier otro campo dentro de una larga lista que podemos consultar en la sección de ayuda de PubMed[1]. También es posible usar los operadores booleanos AND, OR y NOT. Sin embargo, no es necesario emplear las etiquetas ni los operadores, ya que el motor de búsqueda de la página puede crear por sí mismo búsquedas complejas a partir de sólo las palabras clave que introducimos en el campo de búsqueda.

El algoritmo que construye las búsquedas a partir de nuestras palabras clave (*Automatic Term Mapping*) contrasta dichas palabras clave contra diferentes tablas de traducción de términos. En este orden: tabla de traducción de temas, tabla de traducción de revistas, índice de autores e índice de investigadores (colaboradores). Cuando se encuentra una coincidencia para el término o la frase, dicha coincidencia se añade a la búsqueda y no se continúa en la siguiente tabla de traducción.

La tabla de temas relaciona - entre otras cosas - las diferentes formas ortográficas del inglés americano y el británico, formas singulares y plurales, sinónimos, términos fuertemente relacionados, nombres de medicamentos genéricos y sus nombres comerciales, y el vocabulario controlado incluido en el tesauro MeSH (*Medical Subject Headings*).

La tabla de revistas contiene y relaciona el título completo de las revistas, sus abreviaciones y sus números ISSN.

El índice de autores y el índice de investigadores contienen el nombre, iniciales y nombre completo de los autores incluidos en la base de datos.

Primero de todo pruebo a realizar una búsqueda en PubMed (https://pubmed.ncbi.nlm.nih.gov/). La más básica posible y más general, sin ningún filtro, con la palabra clave "endometriosis". El resultado son 29,361 citas, desde 1927 hasta 2021.

[1] https://pubmed.ncbi.nlm.nih.gov/help/#search-tags

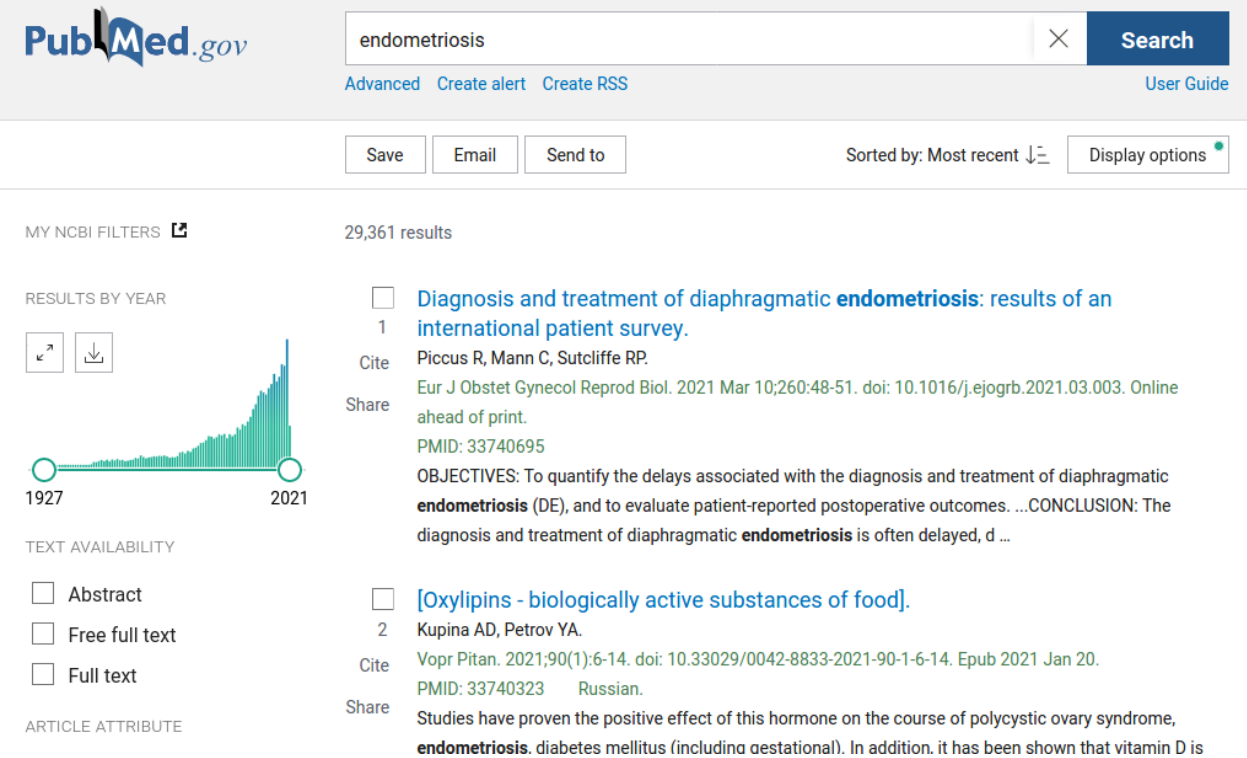


Figura 2: Página de resultados para el término 'endometriosis'. En la parte central se muestran las citas recuperadas por el algoritmo. En el margen izquierdo se nos ofrecen filtros interactivos para refinar la búsqueda.

A través del enlace 'Advanced', que se encuentra en la zona superior izquierda, podemos acceder a un constructor de búsquedas que nos facilita hacer búsquedas avanzadas sin necesidad de conocer todas las etiquetas disponibles. En esa misma página podemos consultar nuestro historial de búsquedas recientes y, en éste, cómo el constructor de búsquedas ha traducido nuestra búsqueda simple ('endometriosis') utilizando las tablas de traducción:

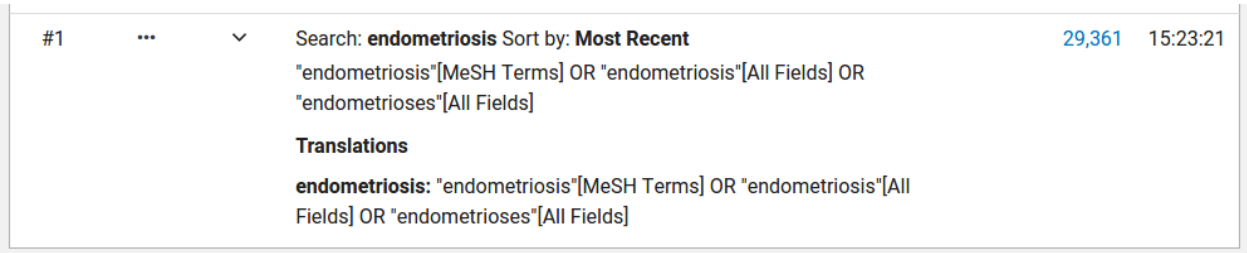
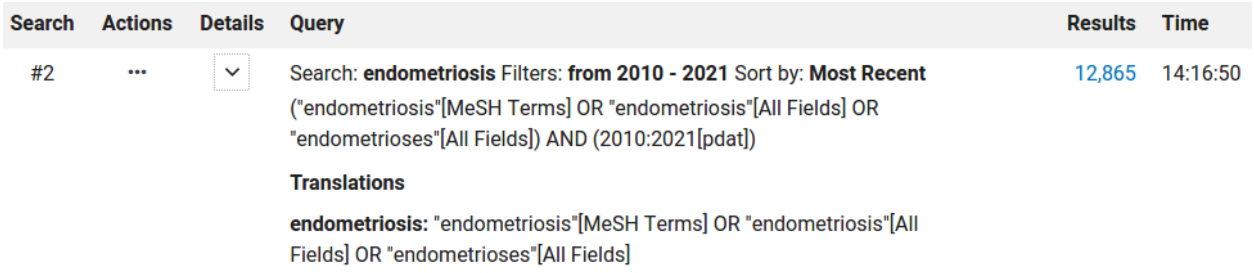


Figura 3: Detalle del historial de búsqueda. De izquierda a derecha muestra la siguiente información: número de la búsqueda en orden cronológico, los términos de búsqueda entrados por el usuario y los términos a los que el algoritmo de mapeo automático los ha traducido, el número de resultados y finalmente la hora a la que se solicitó la búsqueda.

Asimismo si usamos los filtros para, por ejemplo, limitar la búsqueda a los artículos publicados durante los últimos diez años, también podemos consultar la estructura de dicha búsqueda:



Search	Actions	Details	Query	Results	Time
#2	•••	⌄	Search: **endometriosis** Filters: **from 2010 - 2021** Sort by: **Most Recent** ("endometriosis"[MeSH Terms] OR "endometriosis"[All Fields] OR "endometrioses"[All Fields]) AND (2010:2021[pdat]) **Translations** **endometriosis:** "endometriosis"[MeSH Terms] OR "endometriosis"[All Fields] OR "endometrioses"[All Fields]	12,865	14:16:50

Figura 4: Detalle del historial de búsqueda. Misma búsqueda del ejemplo anterior, filtrada para obtener como resultado citas de entre los años 2010 y 2021

2 Exploración de paquetes de minería de textos

2.1 *easyPubMed*

El paquete *easyPubMed* es una interfaz que permite usar R para interactuar con las **Entrez Programming Utilities**, las API[2] públicas que permiten el acceso programático a las bases de datos Entrez (PubMed, PMC, Gene, Nuccore y Protein). Las funciones de este paquete permiten la descarga por lotes de grandes volúmenes de registros, y el procesado básico del resultado de las búsquedas en PubMed(Fantini 2019).

La función principal que usaremos de este paquete es `batch_pubmed_download()`, que permite realizar una búsqueda en PubMed y descargar los resultados en forma de ficheros. Los resultados se pueden descargar en formato XML o TXT en lotes de hasta 5 000 registros. Estos resultados en formato texto conforman los datos sobre los que usaremos las funciones del paquete *pubmed.mineR*.

Como ejemplo, descargaremos los registros correspondientes al término de búsqueda "endometriosis" con fecha de publicación posterior a 2019:

```
query <- "endometriosis AND 2020/01/01:3000/12/31[dp]"

batch_pubmed_download(pubmed_query_string = query,
                      dest_dir = "intermediateData/",
                      dest_file_prefix = "last_endometriosis",
                      format = "abstract",
                      batch_size = 5000
                      )
```

El comporatmiento de la función `batch_pubmed_download()` se puede ajustar mediante diferentes opciones, algunas de las cuales se pueden ver en este ejemplo. Mediante *pubmed_query_string* especificamos los términos con los que se efectuará la búsqueda en PubMed. Puede ser una búsqueda sencilla sólo con los término o, como en el ejemplo, contener etiquetas (ej. [dp], fecha de publicación) y operadores booleanos (ej. AND).

El directorio de destino se puede elegir mediante la opción *dest_dir*, y la opción *dest_file_prefix* nos permite elegir el prefijo que se añadirá a cada uno de los ficheros creados para almacenar los resultados.

Con la opción *batch_size* podemos elegir el número máximo de registros incluidos en cada fichero (entre 1 y 5.000). Por último, la opción *format* nos da la oportunidad de elegir el formato en el que recibiremos los datos. El formato XML es rico en información y permite un procesado posterior más complejo del corpus primario (p.ej. subdividiéndolo por fecha, por autor, u otras opciones). Nosotros sin embargo hemos elegido una de las opciones en formato texto, ya que no necesitaremos realizar ese tipo de procesado del corpus y además resultará en ficheros que, conteniendo el mismo número de registro, ocuparán menos espacio de memoria.

[2]Siglas en inglés de interfaz de programación de aplicaciones (*application programming interface*), que se refiere al conjunto de funciones y protocolos que un programa ofrece para poder ser usado por otro programa diferente.

2.2 *pubmed.mineR*

El paquete *pubmed.mineR*, para el lenguage **R**, se ha desarrollado específicamente para facilitar la minería de textos en el ámbito de la investigación biomédica; concretamente, aplicada a los sumarios (*abstracts*) de artículos incluidos en las citas de la base de datos PubMed. Para este fin, incluye multitud de herramientas que implementan algoritmos de minería de textos o que usan herramientas ya existentes en otros paquetes (Rani, S.Ramachandran, and Shah 2014). En esta sección comentaremos, de entre las muchas funciones contenidas en el paquete, tan sólo aquellas que nos resultarán de utilidad en este trabajo.

En primer lugar, para constituir el **corpus primario**, utilizaremos la función `readabs()` sobre el archivo en formato texto que contiene los registros resultado de nuestra búsqueda previa. El resultado es un objeto tipo S4 con tres *slots* que contienen, respectivamente, la información referente al título de la revista, el texto del sumario y el código PMID[3] del artículo.

```
last_abstracts <- readabs("intermediateData/last_endometriosis01.txt")
```

```
str(last_abstracts, vec.len = 1, nchar.max = 50)
```

```
## Formal class 'Abstracts' [package "pubmed.mineR"] with 3 slots
##   ..@ Journal : chr [1:2276] "1. Eur J Obstet Gynecol Reprod Bi"| __truncated__ ...
##   ..@ Abstract: chr [1:2276] "10.1016/j.ejogrb.2021.02.022. [Ep"| __truncated__ ...
##   ..@ PMID    : num [1:2276] 33756338 ...
```

Disponemos de dos importantes funciones para el **reconomiento de entidades**. La función `gene_atomization()` reconoce los nombres de los genes (en su codificación como símbolo HGNC) y los extrae del del corpus primario además de calcular sus frecuencias.

```
last_genes <- gene_atomization(last_abstracts)
```

```
head(last_genes)
```

```
##      Gene_symbol    Genes
## [1,] "AMH"       "anti-Mullerian hormone"
## [2,] "ARID1A"    "AT-rich interaction domain 1A"
## [3,] "CPP"       "ceruloplasmin pseudogene"
## [4,] "KRAS"      "KRAS proto-oncogene, GTPase"
## [5,] "BDNF"      "brain derived neurotrophic factor"
## [6,] "MALAT1"    "metastasis associated lung adenocarcinoma transcript 1"
##      Freq
## [1,] "128"
## [2,] "58"
## [3,] "56"
## [4,] "32"
## [5,] "30"
## [6,] "26"
```

Por otro lado, la función `word_atomizations()` es más general. Disgrega el texto en palabras y las ordena según su frecuencia. No tiene en cuenta los espacios, la puntuación ni las palabras más comunes del idioma inglés.

```
last_words <- word_atomizations(last_abstracts)
```

[3]PubMed ID; número de identificación único asignado a cada una de las referencias incluidas en la base de datos PubMed.

```
head(last_words)
```

```
##               words Freq
## 13469 endometriosis 7236
## 32774         women 2943
## 24159      patients 2778
## 29603         study 2047
## 23866          pain 1560
## 27215       results 1382
```

Finalmente, la función `tdm_for_lsa()`, a partir de un vector de términos, encuentra la frecuencia de cada término en cada uno de los sumarios del corpus primario. Devuelve una **matriz documento-término** con las frecuencias de los términos buscados en la que cada fila representa uno de los términos y cada columna representa un documento (en este caso, un sumario). Esta matriz se puede usar posteriormente para realizar un análisis semántico latente.

```
tdm <- tdm_for_lsa(last_abstracts,
                   c("age", "gender", "woman", "women", " man ",
                     " men ", "smoking", "relationships", "relation"))
```

```
tdm[, 1:10]
```

```
##               [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
## age              3    0    0    0    2    2    1    1    1     1
## gender           0    0    0    0    0    0    0    0    0     0
## woman            0    0    1    0    0    1    1    0    0     0
## women           10    0    0    0    3    0    0    0    0     3
##  man             0    0    0    0    0    0    0    0    0     0
##  men             0    0    0    0    0    0    0    0    0     0
## smoking          0    0    0    0    0    0    0    0    0     0
## relationships    0    1    0    0    0    0    0    0    0     0
## relation         0    1    0    0    0    0    0    0    0     0
```

3 Generación del corpus primario

El corpus primario para la actividad de minería de texto consistirá en todos los sumarios recuperados de la base de datos PubMed utilizando como término de búsqueda el término "endometriosis". Los registros resultado de la búsqueda se descargarán por lotes en forma de varios archivos de texto, cuyo contenido se reunirá en un único archivo. A partir de este archivo refundido se generará un objeto S4 de clase 'Abstract' conteniendo - de cada registro recuperado - el título, el texto del sumario y el código PMID. Este objeto será lo que consideraremos como corpus primario y la información que tomaremos de partida en los métodos de minería de textos.

3.1 Solicitud de búsqueda y descarga de los registros

Como señalábamos anteriormente, usaremos la función `batch_pubmed_download()` para enviar nuestra búsqueda a la base de datos, recuperar registros y guardarlos en formato texto repartidos en varios ficheros:

```
batch_pubmed_download("endometriosis",
                      dest_dir = "data/",
                      dest_file_prefix = "total_endometriosis",
                      format = "abstract",
                      batch_size = 5000
                      )
```

Previamente, al hacer la búsqueda de prueba en la página de PubMed ya habíamos visto que se recuperan casi 30.000 registros. Lo que significa que el tiempo de descarga será relativamente largo y, al haber especificado que se generaría un archivo por cada 5.000 registros, tendremos como resultado 6 archivos de texto conteniendo todos los registros:

```
[1] "PubMed data batch 1 / 6 downloaded..."
[1] "PubMed data batch 2 / 6 downloaded..."
[1] "PubMed data batch 3 / 6 downloaded..."
[1] "PubMed data batch 4 / 6 downloaded..."
[1] "PubMed data batch 5 / 6 downloaded..."
[1] "PubMed data batch 6 / 6 downloaded..."
[1] "total_endometriosis01.txt" "total_endometriosis02.txt" "total_endometriosis03.txt" "total_endometriosis04.txt"
[5] "total_endometriosis05.txt" "total_endometriosis06.txt"
```

Crearemos un nuevo fichero de texto y copiaremos en él todos los registros, que están repartidos entre los ficheros anteriores:

El resultado será un fichero llamado `todos.txt` conteniendo todos los registros resultado de la búsqueda, en formato TXT. A partir de la información contenida en este fichero generaremos el objeto de clase 'Abstracts' que pueden manipular las funciones del paquete *pubmed.mineR*:

Si examinamos la estructura del objeto:

```
## Formal class 'Abstracts' [package "pubmed.mineR"] with 3 slots
##   ..@ Journal : chr [1:29370] "1. Eur J Obstet Gynecol Reprod Bi"| __truncated__ ...
##   ..@ Abstract: chr [1:29370] "10.1016/j.ejogrb.2021.02.022. [Ep"| __truncated__ ...
##   ..@ PMID    : num [1:29370] 33756338 ...
```

Vemos que consta de 3 *slots*, dos de ellos almacenando datos de tipo cadena de caracteres (título del artículo y su sumario, respectivamente), y un último *slot* de tipo numérico almacenando el código PMID. Es a la información contenida en este objeto a la que aplicaremos los métodos de minería de textos.

4 Preprocesado del corpus primario

El preprocesado que llevaremos a cabo consistirá en desglosar los sumarios en las palabras que los componen y registrar la frecuencia de cada palabra. Para ello usaremos la función `word_atomizations()` del paquete *pubmed.mineR*. Ésta recopilará las palabras del texto y calculará la frecuencia de cada una sin tener en cuenta espacios, signos de puntuación ni palabras muy comunes en inglés.

```
words <- word_atomizations(abstracts)
```

Tabla 1: Las diez palabras más frecuentes en el corpus primario.

Palabras	Frecuencia
endometriosis	66.668
patients	29.232
women	27.229
study	16.110
treatment	15.121
ovarian	13.838
endometrial	12.306
results	12.246
pain	12.019
group	11.417

Entre las palabras más frecuentes encontramos naturalmente la propia palabra clave que hemos usado en la búsqueda (*endometriosis*), términos relacionados con investigación ó tratamiento (*patients, study, treatment, results, group*), con la biología de este trastorno (*women, ovarian, endometrial*) y el síntoma más común (*pain*). El listado completo de palabras, con sus respectivas frecuencias, se puede descargar como fichero de texto desde este enlace.

5 Extracción de genes

Una de las maneras de representar la información contenida en un texto es mediante la extracción de entidades con nombre; como son organizaciones, personas o lugares. En nuestro caso, las entidades de interés son los genes. Partimos de la hipótesis de que los genes que aparecen en los sumarios de artículos acerca de la endometriosis son importantes para este trastorno. Así pues, extraeremos un listado de los mismos para, más adelante, recuperar información a partir de los términos de ontología génica que estos genes tengan en común. Para realizar la extracción de términos usaremos la función `gene_atomization()` del paquete *pubmed.minerR* de R. Esta función reconoce, y recupera de los sumarios, los símbolos aprovados por el HGNC[4] para representar genes concretos. Esta función devuelve el símbolo del gen, su nombre largo y su frecuencia en el corpus.

```
genes <- gene_atomization(abstracts)
```

Tabla 2: Los diez genes más frecuentes en el corpus primario.

Símbolo	Nombre largo	Frecuencia
AMH	anti-Mullerian hormone	721
CPP	ceruloplasmin pseudogene	445
ARID1A	AT-rich interaction domain 1A	273
PIP	prolactin induced protein	210
PTEN	phosphatase and tensin homolog	209
HOXA10	homeobox A10	194
EGF	epidermal growth factor	158
MIF	macrophage migration inhibitory factor	156
GSTM1	glutathione S-transferase mu 1	154
NGF	nerve growth factor	154

6 Referencias

Fantini, Damiano. 2019. *EasyPubMed: Search and Retrieve Scientific Publication Records from Pubmed*. https://CRAN.R-project.org/package=easyPubMed.

National Library of Medicine. 2021a. "About - PubMed." Accessed March 9. https://pubmed.ncbi.nlm.nih.gov/about/.

———. 2021b. "MEDLINE Overview." Accessed March 9. https://www.nlm.nih.gov/medline/medline_overview.html.

Rani, Jyoti, S.Ramachandran, and Ab. Rauf Shah. 2014. *Pubmed.mineR: An R Package for Text Mining of Pubmed Abstracts*. https://CRAN.R-project.org/package=pubmed.mineR.

[4]HUGO Gene Nomenclature Commitee (Comité de Nomenclatura de Genes de la HUGO), https://www.genenames.org/