Aplicación web para el estudio del estado del arte en temas biomédicos aplicando técnicas de minería de textos

María Dolores Herráiz Lablanca
Master Universitario en Bioinformática y Bioestadística
Computación e Inteligencia Artificial en problemas biológicos y clínicos

Consultora: *Romina Astrid Rebrij*
Profesor responsable de la asignatura: *David Merino Arranz*

Fecha Entrega: 3 de enero de 2019

FICHA DEL TRABAJO FINAL

Título del trabajo:	*Aplicación web para el estudio del estado del arte en temas biomédicos aplicando técnicas de minería de textos*
Nombre del autor:	*María Dolores Herráiz Lablanca*
Nombre de la consultora:	*Romina Astrid Rebrij*
Nombre del PRA:	*David Merino Arranz*
Fecha de entrega (mm/aaaa):	03/2019
Titulación:	*Master Universitario en Bioinformática y Bioestadística*
Área del Trabajo Final:	*Computación e Inteligencia Artificial en problemas biológicos y clínicos*
Idioma del trabajo:	*Español*
Palabras clave	*Data-minig, radioterapia, PubMed*

Resumen del Trabajo (máximo 250 palabras): *Con la finalidad, contexto de aplicación, metodología, resultados i conclusiones del trabajo.*

Cualquier trabajo de investigación implica una búsqueda bibliográfica exhaustiva, a fin de identificar y localizar bibliografía sobre el tema en cuestión. Realizar manualmente el análisis de toda la información obtenida es una tarea tediosa y repetitiva que requiere de mucho tiempo y esfuerzo. El uso de una aplicación informática puede aligerar esta carga y facilitar el trabajo del investigador.

Con este objetivo, el presente trabajo ha consistido en la elaboración de una aplicación web, utilizando el software libre R y distintos paquetes de minería de textos, que permite tabular la información obtenida tras una búsqueda en PubMed y presentarla en forma gráfica, lo que agiliza su análisis, al ser realizado con mayor rapidez y eficacia. Su estructura en bloques la hace de sencillo manejo, y la aplicación de métodos de minería de texto y estadística multivariante, permite la identificación de información útil e importante de los datos no estructurados de las sinopsis. Al poder compartirse en internet, la aplicación web puede ser utilizada por cualquier usuario, incluso aquellos sin conocimientos en lenguajes de programación.

La aplicación está orientada al estudio de toxicidades en tratamientos de radioterapia para distintas localizaciones y se ha aplicado al tema específico "toxicidades en tratamientos de radioterapia de próstata".

Abstract (in English, 250 words or less):

Any research work involves a thorough bibliographic search, in order to identify and locate bibliography on the subject in question. A manually performed analysis of all available information is a tedious and repetitive task, challenging and time-consuming. Using a computer application can alleviate the burden and simplify the researcher's work.

With this purpose, the present work consisted of the development of a web application, using the GPL software R and different text mining packages, allowing to tabulate the obtained information from a PubMed search and showing it graphically. This speed its analysis up, being performed more quickly and effective. Its structure in blocks, makes it use friendly, and the application of text mining methods and multivariate statistics, allows the extraction of useful and important information from the unstructured data of the abstracts. Sharing the application on the internet, makes it available for any user, even those without knowledge in programming languages.

The application is aimed to the study of toxicities in radiotherapy treatments for different locations and has been applied to the specific topic "toxicities in prostate radiotherapy treatments".

Índice

Lista de figuras

1. Introducción

1.1 Contexto y justificación del Trabajo

Al inicio de cualquier trabajo de investigación se debe realizar una búsqueda bibliográfica exhaustiva identificando y localizando bibliografía sobre un determinado tema.

El portal PubMed [1] nos permite el acceso a MEDLINE y a otras bases de datos desarrolladas por la National Library of Medicine, PreMEDLINE, Genbak y Complete Genoma. Realizar un análisis manual de toda la información obtenida tras la búsqueda bibliográfica, requiere de mucho tiempo. Para optimizar el tiempo empleado en esta tarea y facilitar el trabajo del investigador, se puede hacer uso de técnicas de minería de textos y análisis multivariante [2,3,4,5].

R es un software libre, y uno de los lenguajes más utilizados en investigación por la comunidad estadística, siendo además muy popular en el campo de la minería de datos [6]. Han sido muchos los paquetes de R que se han desarrollado para minería de textos en general como *tm* [7], *quanteda* [8], para clasificación de textos [9] y algunos específicos para los textos extraídos de búsquedas en PudMed como *pubmed.mineR* [10], *RISmed* [11] y *bibliometrix* [12]. Hay muchas publicaciones sobre minería de textos, en la que se comparan diversos paquetes de R [13] o se utilizan para minería de textos biológicos [14,15,16,17], pero no se ha encontrado ninguna aplicación web en la que utilice minería de textos para hacer una búsqueda bibliográfica general de forma interactiva. En la actualidad, los mismos desarrolladores del paquete bibliometrix han desarrollado *biblioshiny*[18], una aplicación para búsqueda bibliográfica, pero hay que tener instalado R para poder acceder a la misma.

Se ha encontrado literatura sobre algunos paquetes de R o programas desarrollados para hacer minería de textos en Medline, pero no son software libre o el enlace indicado en la publicación para la adquisición del paquete da problemas [19,20].

Con la aplicación web desarrollada en este TFM, en la que se utilizan distintos paquetes de R para minería de textos, se puede extraer información de los resúmenes de forma automática reduciendo así el tiempo de procesado de la información.

1.2 Objetivos del Trabajo

El TFM consistió en la elaboración de una aplicación web que permitiera estudiar el estado del arte de un tema perteneciente al ámbito biomédico, aplicando técnicas de minería de texto, en este caso, ***toxicidades en distintas localizaciones tras recibir tratamientos de radioterapia***. La aplicación web se aplica de forma más específica a tratamientos de próstata.

El objetivo general:

1. Elaborar una aplicación web para extraer y evaluar la información obtenida tras una búsqueda bibliográfica.

Y los objetivos específicos:

1. Identificar bibliografía por tema y por año.

2. Identificar autores más frecuentes por tema.

3. Identificar artículos mas influyentes.

4. Establecer relaciones entre temas de un ámbito específico y metodologías aplicadas.

1.3 Enfoque y método seguido

Para extraer información de Pubmed, se podría elaborar un informe dinámico en Markdown o una aplicación web. Se ha elegido la aplicación web, por ser más interactiva, y más parametrizable para ser utilizada en búsquedas generales. Al poder compartirse en internet, la aplicación web puede ser utilizada por otros usuarios de PubMed que deseen extraer de manera automática información de los abstracst, pero, que no tengan conocimientos en lenguajes de programación.

El producto desarrollado es nuevo, y se ha elaborado utilizando el software R, en especial el paquete *bibliometrix*, del que se han adaptado algunas funciones.

El método utilizado ha sido el siguiente:

1. Se ha realizado la búsqueda en PubMed. Esta búsqueda se hace de dos maneras distintas: con EUtilsSummary() del paquete RISmed, y directamente en la página web https://www.ncbi.nlm.nih.gov/pubmed/

2. Se ha creado un dataframe con la información obtenida con pubmed2df() del paquete *bibliometrix* y un fichero al realizar la búsqueda en la página web https://www.ncbi.nlm.nih.gov/pubmed/

3. Se ha utilizado el paquete *bibliometrix* para visualizar información bibliográfica, y los paquetes como *pubmed.mineR* y *tm* para buscar relaciones de términos en el cuerpo del abstract.

4. Se aplica al tema '***Toxicidades en tratamientos de próstata para distintos tipos de tratamiento***.

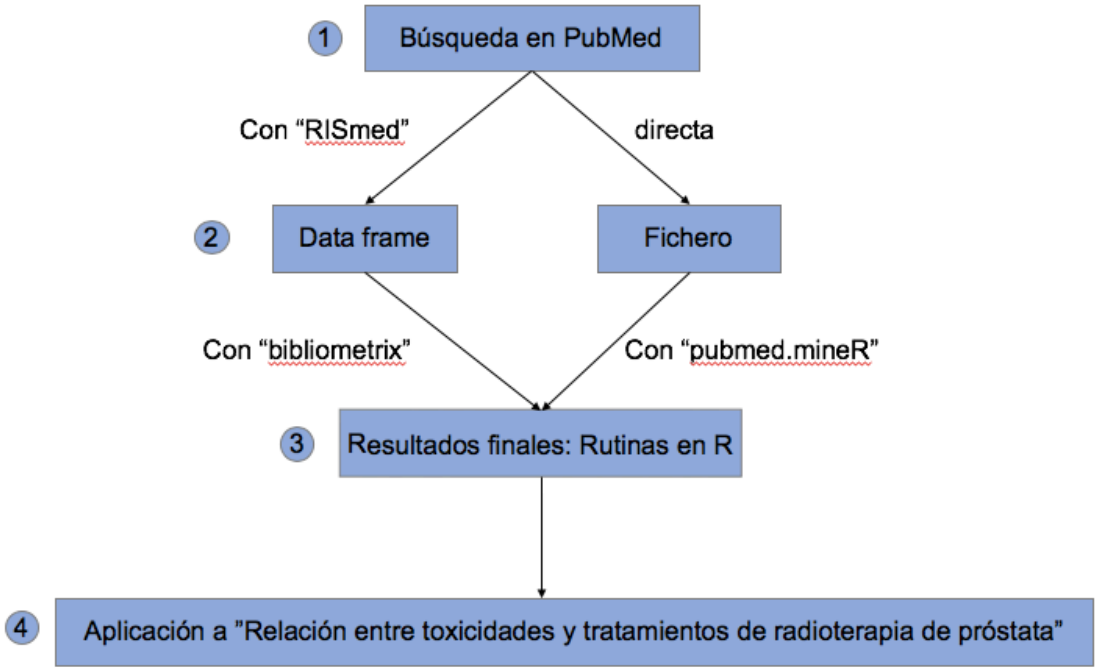

Figura 1. *Metodología utilizada*

1.4 Planificación del Trabajo

El trabajo se estructuró de acuerdo con la temporalización de las distintas entregas de las PECs, dividiéndose en los apartados siguientes:

- **Definición de los contenidos del trabajo**: en esta etapa se definió la temática del trabajo justificando su interés y relevancia. En la propuesta se incluían: el título del proyecto, palabras clave, temática escogida, problemática a resolver, objetivos y bibliografía utilizada.

- **Plan de trabajo**: En este periodo de tiempo se elaboró un documento consensuado con la consultora. Se definieron las líneas generales del proyecto y su enfoque, los objetivos, se estableció una temporalización apropiada del proyecto y se definieron los riesgos y acciones de mitigación.

- **Desarrollo del trabajo Fase 1**: en el periodo comprendido entre mediados de octubre y mediados de noviembre de 2018, se eligieron los paquetes de R para minería de texto tanto específicos para PubMed como no específicos, se escribieron las rutinas en R y se instaló el paquete *shiny*. El hito en esta fase fue tener las rutinas escritas en R antes de la entrega de la PEC 2 el 19.11.2018

- **Desarrollo del trabajo Fase 2**: en el periodo que comprendió de mediados de noviembre a mediados de diciembre de 2018, se diseñó la plataforma web adaptando las rutinas escritas en R al formato utilizado por el paquete *shiny*. Tras colgar la aplicación en el servidor shiny se depuraron los errores detectados durante la fase de prueba. El hito identificado en esta fase era tener la aplicación web colgada en el servidor shiny antes de la entrega de la PEC 3 el 17.12.2018

- **Redacción de la memoria**: se utilizó la plantilla suministrada por la Universidad para redactar la memoria.

- **Elaboración de la presentación**: se realizó un video en el que la alumna explicaba los aspectos más relevantes del proyecto.

En la Tabla 1, se presentan las tareas realizadas:

Nombre de la tarea	Fecha inicial	Fecha final	Duración (dias)
TFM	**19/09/2018**	**23/01/2019**	**126**
Definición de los contenidos del trabajo	**19/09/2018**	**01/10/2018**	**12**
Plan de trabajo	**02/10/2018**	**15/10/2018**	**13**
Desarrollo del trabajo – Fase 1	**16/10/2018**	**19/11/2018**	**34**
Evaluación distintos paquetes de R para minería de datos específicos de PubMed	16/10/2018	19/11/2018	34
Elección de distintos paquetes de R para minería de datos no específicos de PubMed	16/10/2018	19/11/2018	34
Elección de los paquetes que se usarán en la aplicación web	16/10/2018	19/11/2018	34
Escritura de las rutinas en R para extraer información de la búsqueda bibliográfica	16/10/2018	19/11/2018	34
Instalación de shiny	16/10/2018	19/11/2018	34
Desarrollo del trabajo – Fase 2	**20/11/2018**	**17/12/2018**	**27**
Aprendizaje del manejo de shiny con tutoriales y programas sencillos	20/11/2018	17/12/2018	27
Diseño de la aplicación web	20/11/2018	17/12/2018	27
Adaptación de las rutinas escritas en R a la aplicación web	20/11/2018	17/12/2018	27
Aplicación al tema “Relación entre tratamientos y toxicidades en Radioterapia de próstata”	20/11/2018	17/12/2018	27
Redacción de la memoria	**18/12/2018**	**02/01/2019**	**15**
Desarrollo de la presentación	**03/01/2019**	**10/01/2019**	**7**
Defensa pública	**14/01/2019**	**23/01/2019**	**9**

Tabla 1. *Tareas realizadas*

En la Figura 2 se muestra el diagrama de Gantt con la temporalización:

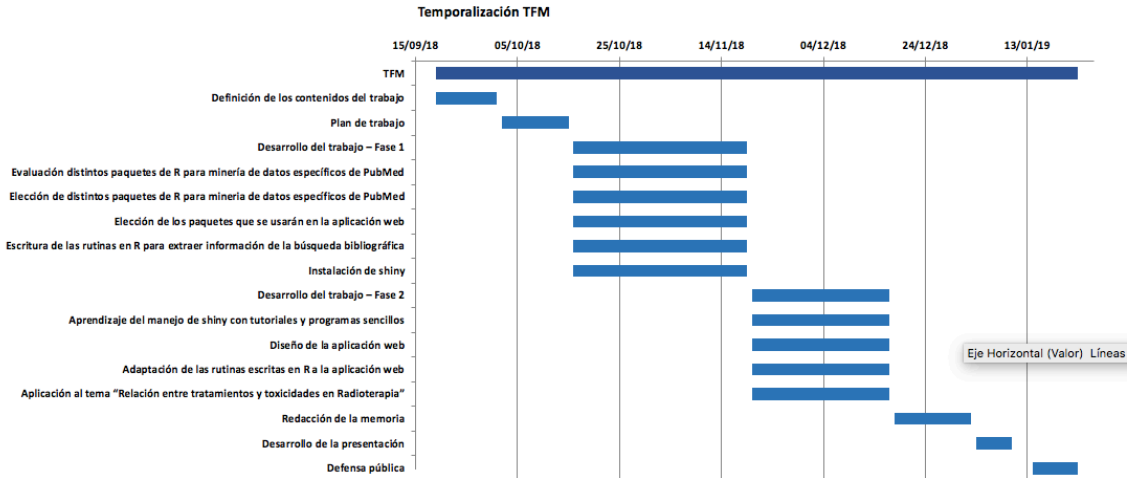

Figura 2. *Diagrama Gantt*

1.5 Breve sumario de productos obtenidos

El producto obtenido es la aplicación web que está colgada en:

https://malea.shinyapps.io/pec_3_app/

Además, se enviará una comunicación al 6ª Congreso conjunto de SEFM-SEPR que tendrá lugar del 11 al 14 de junio en Burgos.

1.6 Breve descripción de los otros capítulos de la memoria

En los capítulos siguientes, se presenta la introducción teórica, metodología utilizada, resultados obtenidos, y conclusiones.

En el capítulo 2, se habla de PubMed y su uso.

En los capítulos 3 y 4, se hace una introducción a dos conceptos fundamentales de la minería de textos, como son el análisis de semántica (LSA) y la clasificación jerárquica en clusters.

En el capítulo 5, se concretan los materiales y métodos aplicados en el presente trabajo.

En el capítulo 6, se presentan los resultados obtenidos al utilizar la plataforma para estudiar las relaciones entre toxicidades y tratamientos en el caso de radioterapia de próstata.

En el capítulo 7, se presentan las conclusiones, con una reflexión crítica sobre el logro de los objetivos y el seguimiento de la planificación, así como las líneas de trabajo futuro.

En el capítulo 8, se definen los términos y acrónimos más relevantes utilizados dentro de la memoria.

En el capítulo 9, se presenta una lista numerada de las referencias bibliográficas utilizadas dentro de la memoria.

Finalmente, en el Anexo se adjunta el código fuente de la aplicación.

2. PubMed

PubMed es un motor de búsqueda desarrollado por *el* ***National Center for Biotechnology Information*** (**NCBI**) en la ***National Library of Medicine*** (**NLM**) de Estados Unidos, que permite el libre acceso a las bases de datos compiladas por la **NLM**: **MEDLINE**, y otras.

MEDLINE es la base de datos referencial más importante de la **NLM**, provee del acceso a más de 12 millones de referencias bibliográficas de artículos de 4600 revistas biomédicas publicadas en Estados Unidos y 70 países desde el año 1966 hasta hoy. Abarca los campos de la medicina, enfermería, odontología, veterinaria, salud pública, y ciencias preclínicas. Es un recurso de consulta indispensable para todo el personal de salud. MEDLINE es el principal componente de la base de datos de **PubMed**, a la que se puede acceder a través de Internet. Además de proveer acceso a MEDLINE, PubMed permite recuperar referencias que están fuera del área temática de cobertura (por ejemplo: astrofísica o geología), de ciertas revistas indizadas en MEDLINE. También en PubMed podemos encontrar citas anteriores a 1966 y artículos de algunas revistas que envían su texto completo al PubMedCentral y reciben una revisión cualitativa por la NLM. Medline es accesible vía internet desde NLM, pero algunas plataformas y portales científicos de pago también lo ofrecen, es el caso de desde Ovid, Silver Platter, y desde el 2009 WOK.

Todas las citas del MEDLINE tienen asignados términos **MeSH** o ***Medical Subject Headings,*** un vocabulario controlado de términos biomédicos que representan el contenido de cada artículo que se ingresa a la base de datos **MEDLINE**. El **MeSH** contiene cerca de 19.000 expresiones que son revisadas anualmente y actualizadas de acuerdo con los avances de la práctica médica y de la terminología.

PMID, acrónimo de «PubMed Identifier» o «PubMed Unique Identifier», es un número único asignado a cada cita de un artículo de revistas biomédicas y de ciencias de la vida que recoge PubMed. Este registro es de la Biblioteca Nacional de Medicina de los Estados Unidos (MEDLINE).

En MEDLINE cada campo de un registro bibliográfico se identifica mediante una etiqueta de dos o más letras (calificadores de campo), que podemos añadir a continuación de cada término entre corchetes: ej.: herraiz[au].

Tanto los términos de búsqueda como los calificadores de campo, no importa si se escriben en mayúscula o minúsculaej.: radiotherapy [mh] = Radiotherapy[mh] = RADIOTHERAPY [mh]

A continuación, se muestra una tabla con los calificadores de campos o etiquetas:

Field	Abbreviation
Abstract	(AB)
Copyright Information	(CI)
Affiliation	(AD)
Investigator Affiliation	(IRAD)
Article Identifier	(AID)
Author	(AU)
Author Identifier	(AUID)
Full Author	(FAU)
Book Title	(BTI)
Collection Title	(CTI)
Comments/Corrections	
Conflict of Interest Statement	(COIS)
Corporate Author	(CN)
Create Date	(CRDT)
Date Completed	(DCOM)
Date Created	(DA)
Date Last Revised	(LR)
Date of Electronic Publication	(DEP)
Date of Publication	(DP)
Edition	(EN)
Editor and Full Editor Name	(ED) (FED)
Entrez Date	(EDAT)

Field	Abbreviation
Gene Symbol	(GS)
General Note	(GN)
Grant Number	(GR)
Investigator Name and Full Investigator Name	(IR) (FIR)
ISBN	(ISBN)
ISSN	(IS)
Issue	(IP)
Journal Title Abbreviation	(TA)
Journal Title	(JT)
Language	(LA)
Location Identifier	(LID)
Manuscript Identifier	(MID)
MeSH Date	(MHDA)
MeSH Terms	(MH)
NLM Unique ID	(JID)
Number of References	(RF)
Other Abstract	(OAB)
Other Copyright Information	(OCI)
Other ID	(OID)
Other Term	(OT)
Other Term Owner	(OTO)
Owner	(OWN)

Field	Abbreviation
Pagination	(PG)
Personal Name as Subject	(PS)
Full Personal Name as Subject	(FPS)
Place of Publication	(PL)
Publication History Status	(PHST)
Publication Status	(PST)
Publication Type	(PT)
Publishing Model	(PUBM)
PubMed Central Identifier	(PMC)
PubMed Central Release	(PMCR)
PubMed Unique Identifier	(PMID)
Registry Number/EC Number	(RN)
Substance Name	(NM)
Secondary Source ID	(SI)
Source	(SO)
Space Flight Mission	(SFM)
Status	(STAT)
Subset	(SB)
Title	(TI)
Transliterated Title	(TT)
Volume	(VI)
Volume Title	(VTI)

***Figura 3**. Calificadores de campos o etiquetas [21]*

3. Minería de textos mediante análisis de semántica latente

Una de las principales técnicas de minería de textos es el análisis de semántica latente (LSA), que supone que ciertas palabras aparentemente independientes están relacionadas por temas subyacentes no observados. Por ejemplo, las palabras “alumno” y “aula” pueden considerarse pueden considerarse expresiones superficiales de un tema latente más relevante como es “escuela”. Fue desarrollada por Landauer y Dumais [22] y permite generar conocimiento global indirectamente a partir de los datos de coocurrencia local en un gran cuerpo de texto representativo. LSA no utiliza conocimientos previos de similitud lingüística o perceptiva, se basa únicamente en un método de aprendizaje matemático general que logra poderosos efectos inductivos al extraer el número correcto de dimensiones para representar objetos y contextos.

Pasos seguidos:

1. El LSA comienza procesando un texto, normalmente de grandes dimensiones, contiendo miles de palabras, párrafos y frases llamado ***corpus*.**

2. Se construye la **Matriz término-documento**, una matriz de frecuencias cuyas filas corresponden a los distintos términos del corpus y en cuyas columnas aparecen los distintos documentos, conteniendo el número de veces que cada término aparece en el documento.

3. A continuación, se realiza una **ponderación** con el fin de restar importancia a los términos muy frecuentes ya que en cualquier texto aparecen reiteradas veces artículos y determinantes que no aportan información relevante; y aumentarla a los menos frecuentes. Las palabras excesivamente frecuentes no nos sirven para seleccionar bien la información relevante del párrafo, pero, las que aparecen de forma moderada sí.

4. Se aplica una técnica matemática llamada **descomposición de valores singulares** (SVD) [23] para reducir el número de filas sin perder información importante.

5. Luego se calcula el coseno del ángulo entre los dos vectores (o el producto de puntos entre las normalizaciones de los dos vectores) formados por dos filas cualquiera. Los valores cercanos a 1 representan palabras muy similares, mientras que los valores cercanos a 0 representan palabras muy diferentes.

4. Métodos Jerárquicos de Análisis Cluster

El análisis de cluster es una técnica cuya idea básica es agrupar un conjunto de observaciones en un número dado de clusters o grupos. Este agrupamiento se basa en la idea de distancia o similitud entre las observaciones.
La obtención de dichos clusters depende del criterio o distancia considerados; así, por ejemplo, una baraja de cartas españolas se podría dividir de distintos modos: en cuatro clusters (los cuatro palos), en ocho clusters (los cuatro palos y según sean figuras o números), en dos clusters (figuras y números). Es decir, todo depende de lo que consideremos como similar.
El número de combinaciones posibles de grupos y de los elementos que los integran se hace intratable desde el punto de vista computacional, aún con un número escaso de elementos. Se hace necesario, pues, encontrar métodos o algoritmos para calcular el número de clusters y componentes más aceptable, aunque no sea el óptimo absoluto.
Los métodos jerárquicos se subdividen en aglomerativos y disociativos:

- Los **metodos aglomerativos**, también conocidos como ascendentes, comienzan el análisis con tantos grupos como individuos haya. A partir de estas unidades iniciales se van formando grupos, de forma ascendente, hasta que al final del proceso todos los casos tratados están englobados en un mismo conglomerado.

- Los **métodos disociativos**, también llamados descendentes, constituyen el proceso inverso al anterior. Comienzan con un conglomerado que engloba a todos los casos tratados y, a partir de este grupo inicial, a través de sucesivas divisiones, se van formando grupos cada vez más pequeños. Al final del proceso se tienen tantas agrupaciones como casos han sido tratados.

En cualquier caso, de ambos métodos se deriva un **dendograma**, que es un gráfico que ilustra cómo se van haciendo las subdivisiones o los agrupamientos, etapa a etapa.

Consideramos aquí los métodos aglomerativos con diferentes ***métodos de unión*** (linkage methods). Los más importantes son:

- *Mínima distancia* o vecino más próximo.
- *Máxima distancia* o vecino más lejano.
- Distancia media (*average distance*).
- *De Ward* es un procedimiento jerárquico en el cual, en cada etapa, se unen los dos clusters para los cuales se tenga el menor incremento en el valor total de la suma de los cuadrados de las diferencias, dentro de cada cluster, de cada individuo al centroide del cluster. Éste es el método utilizado para realizar la clasificación de los resúmenes en el presente trabajo.

5. Materiales y métodos.

5.1. Elección de librerías

Para la elección de los paquetes de R a instalar, se han comparado distintas librerías de R para minería de textos tanto específicas de PubMed como no específicas, además de otras para realizar gráficos y tablas.

5.1.1. Paquetes para mineria en PubMed

- **pubmed.mineR:** Con este paquete es necesario que los abstracts se obtengan previamente utilizando el motor de búsqueda PubMed, guardando localmente los resultados de la búsqueda como se muestra en la Figura 4.

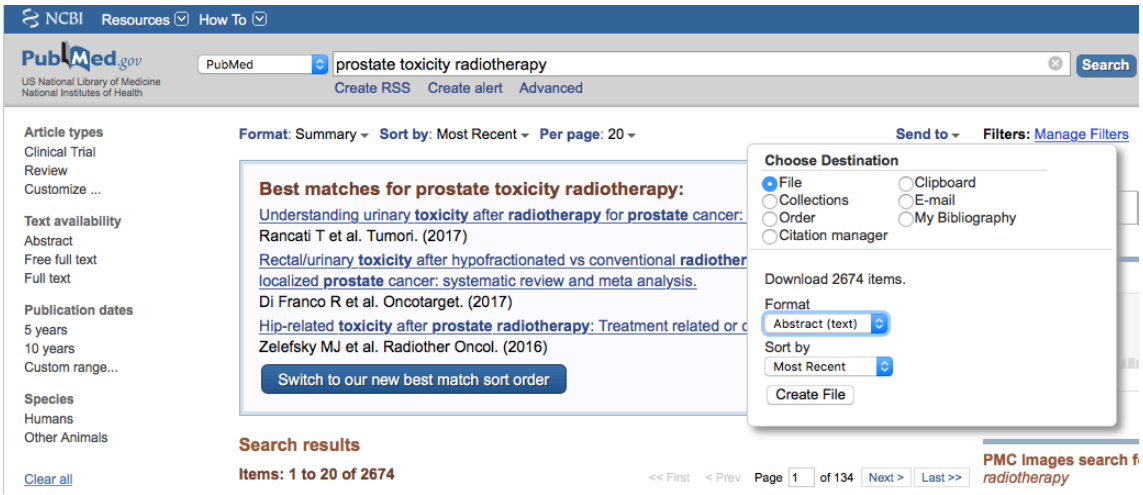

Figura 4. *Ejemplo descarga de abstracts*

Tras realizar la búsqueda, los abstracts se leen con la función readabs(). En la Tabla 2, se muestra una comparativa entre los tiempos de ejecución al realizar la búsqueda y lectura de los abstracts para los distintos paquetes de R. En el caso de esta librería lo que se calcula es el tiempo de lectura del archivo de abstracts.

- **RISmed:** El paquete RISmed permite utilizar la función EUtilsSummary () para realizar búsquedas en PubMed.

- **easyPubMed:** easyPubMed es una interfaz R para consultar NCBI Entrez y recuperar los registros de PubMed en formato XML o TXT, que se pueden descargar y guardar en dicho formato. El paquete es adecuado para la descarga por lotes de grandes volúmenes de registros (a través de la función batch_pubmed_download ()) y también incluye un conjunto de funciones para realizar el procesamiento básico de la salida de la consulta de PubMed. La integridad de los datos se aplica durante la descarga de datos, lo que permite recuperar y guardar una gran cantidad de registros sin esfuerzo. Los registros de PubMed se pueden procesar para extraer información específica de la publicación y del autor.

- **bibliometrix:** Se utiliza la función EUtilsSummary() de RISmed para realizar la búsqueda. Lo interesante de este paquete es que utilizando la función pubmed2df(), se obtiene un data frame con 22 variables entre las que tenemos:
    - "AU": Autor
    - "TI": Título
    - "AB": Resumen
    - "PY": Año de publicación
    - "MESH": Términos MeSH
    - "TC": Veces que ha sido citado
    - …

  Manejar y extraer la información guardada en modo de dataframe es muy sencillo, aunque como vemos en la Tabla 2, la búsqueda con bibliometrix es costosa en tiempo de ejecución (el tiempo corresponde casi en su totalidad al uso de la función pubmed2df()).

Paquete	Tiempo (s.)
pubmed.mineR	1.3
RISmed	1.0
easyPubMed	6.4
bibliometrix	2984.0

Tabla 2. *Tiempo de ejecución de las rutinas de búsqueda y grabado de abstracts.*

Bibliometrix tarda hasta 3000 veces más que los otros paquetes en buscar y guardar los resúmenes.

Respecto a los paquetes de R para minería de textos específicos de PubMed, el más completo es *bibliometrix*, con el inconveniente del tiempo de ejecución de la función pubmed2df() para resultados de búsquedas con más de 200 abstracts. Se ha realizado la busqueda y guardado el data frame así la aplicación web solo tiene que leerlo que es mucho más rápido que generarlo. Los tres paquetes utilizados son: *pubmed.mineR*, *RISMed*, y *Bibliometrix*.

5.1.2. Paquetes para minería no específicos de pubMed

- **tm:** Éste es uno de los paquetes más usado para minería de textos. Aplicamos la función tm_map() al corpus para la eliminación de: números, puntuación, palabras vacías (palabras más frecuentes del idioma inglés como artículos, etc.) y para lematizar, es decir, mantener solo la raíz de las palabras. La función DocumentTermMatrix(), se utiliza para calcular la matriz término-documento.

- t**extmineR:** Es una ayuda para la minería de texto en R, con una sintaxis que debe ser familiar para los usuarios de R con experiencia, además tiene funcionalidad adicional para el análisis y diagnóstico de *topic models*. En el presente trabajo, se utiliza la función CreateDtm() y TermDocFreq() para la generación de la matriz término-documento.

- **wordcloud:** paquete para realizar nubes de palabras.

- **lsa:** paquete para hacer análisis semántico latente.

5.1.3. Paquetes para clusters

- **factoextra:** Se utiliza la función fviz_cluster() que proporciona una visualización elegante basada en ggplot2 de los métodos de clasificación en clusters.

- **stats:** La función hclust(), se usa para realizar la agrupación en clusters jerárquica y cutree() para separar los conglomerados resultado de aplicar la función hclust().

5.1.4. Paquetes para visualizar gráficos

- **ggplot2:** No se utiliza directamente este paquete, pero como se ha comentado antes factoextra proporciona una visualización elegante basada en esta librería, y bibliometrix la usa también para realizar gráficos.

- **gplots:** Este paquete se utiliza para realizar representaciones gráficas. Hemos utilizado la función heatmap.2() de este paquete en una de las rutinas.

- **Igraph:** Los principales objetivos de la biblioteca igraph es proporcionar un conjunto de clases de datos y funciones para la implementación sencilla de algoritmos de gráficos y el manejo rápido de gráficos grandes, con millones de vértices y bordes. Utilizamos la función cluster_walktrap() que trata de encontrar subgrafos densamente conectados, también llamados comunidades en un gráfico a través de caminatas aleatorias. La idea es que los paseos aleatorios cortos tienden a permanecer en la misma comunidad.

5.1.5. Paquetes para realizar tablas

- **DT:** El paquete DT proporciona una interfaz R a la biblioteca de JavaScript DataTables. Los objetos de datos R (matrices o marcos de datos) se pueden mostrar como tablas en páginas HTML, y DataTables proporciona filtrado, paginación, clasificación y muchas otras características en las tablas.

- **kableExtra:** Es un paquete para realizar tablas muy elegantes. En la Fase 1, en la que se trabajaba en la elaboración de las rutinas se utilizó este paquete, pero, al acoplar las rutinas al paquete *shiny* para configurar la aplicación se dejó de utilizar y se usó el paquete *DT* para elaborar todas las tablas que aparecen en la aplicación.

5.2. Elaboración de rutinas

Se utiliza el paquete *bibliometrix* para elaborar la mayor parte de las rutinas que son ejecutadas al seleccionar la pestaña "Generalidades", que al ser de código abierto [24] permite ser modificado y las rutinas a las necesidades del presente proyecto.
Se parte de un dataframe con 22 variables, de las cuales las más interesantes son:

- AU: Autor
- TI: Título
- AB:Abstract
- PY:Año de publicación
- MeSH: Medical Subject Headings
- TC: Número de citas que tiene el artículo
- AU_CO: País de afiliación de cada uno de los coautores

5.2.1. Producción científica anual.

Con la función plot_Prod.bibliometrix(), se puede ver la producción anual. Esta función no es más que una parte de la función plot.bibliometrix() del paquete *bibliometrix*, modificada.

5.2.2. Lista de Artículos.

Se utiliza la función datatable() del paquete DT para mostrar la lista de artículos que hablan sobre el tema elegido. Se muestra en la primera columna el código PMID, y al seleccionarlo, se accede al abstrtact en PubMed en una nueva pestaña. En la segunda columna se muestra el título y la tercera y cuarta, año de publicación y el número veces que ha sido citado. Seleccionando la columna "*año"* o "*citas"* se puede ordenar la tabla en modo ascendente o descendente.

5.2.3. Autores más prolíficos.

Se utiliza la función plot_Autores(), que es una función escrita a partir de la función plot.bibliometrix() del paquete bibliometrix, para visualizar un histograma horizontal en el que se muestran a los 10 autores más prolíficos. En el eje Y aparecen el nombre de los autores y en las X el número de publicaciones totales.

5.2.4. Factor de Dominancia.

Se ha modificado la función dominance() de *bibliometrix* dando como resultado la función dominanceL() para calcular el factor de dominancia (DF) y elaborar una tabla con el factor de dominancia en la primera columna definido como:

$$DF = \frac{Nº\ de\ Artículos\ como\ Autor\ Principal}{Nº\ de\ Artículos\ como\ coautor}$$

En la segunda y tercera columna, se muestran el número de artículos escritos como coautor y autor principal.

En la cuarta y quinta, se muestran el ranking por número de artículos y por DF.

5.2.5. Artículos de los autores más prolíficos.

En el menú de la izquierda aparece una lista con los autores más prolíficos y al elegir uno, se obtiene la lista de los artículos escritos por dicho autor.

5.2.6. Publicaciones de paises por año.

En este parte, se utiliza la función Heatmap.2() del paquete *gplots* para visualizar la evolución temporal de la producción anual por país.

5.2.7. Paises más productivos.

Se muestra un histograma horizontal de los países que más publican, tanto solos (en color azul) como en colaboración con otros países (en color rojo). En este caso se ha utilizado la función plot_paises.bibliometrix(), también basada en una función del paquete bibliometrix.

5.2.8. Colaboraciones entre paises.

Se utiliza la función metaTagExtraction() que extrae información de los metadatos, para conocer la afiliación de cada uno de los coautores, biblioNetwork(), y por último la función networkPlot() para generar un gráfico circular, en el que se conectan lo países que colaboran juntos.

5.2.9. Dosis utilizadas.

En el estudio de toxicidades para distintas localizaciones, es importante saber la dosis de radiación administrada y el fraccionamiento usado en cada tratamiento ya que, los efectos adversos, es decir, las toxicidades, dependen tanto de la dosis total como de la dosis por fracción. Es interesante hacer la búsqueda por años, ya que los fraccionamientos y dosis van evolucionando con el tiempo. Debido al avance tecnológico, la aplicación de los tratamientos es cada vez más precisa, siendo posible administrar en la actualidad, más dosis y más dosis por fracción que hace unos años.

Partiendo del dataframe, se hace un cribado de las sinopsis que contengan la palabra “Gy”. El Gray (símbolo Gy) es una unidad derivada del Sistema internacional de Unidades que mide la dosis absorbida procedente de radiaciones ionizantes por un determinado material. Un gray es equivalente a la absorción de un julio de energía por un kilogramo de masa de material irradiado. Después, se separan los resúmenes en frases, seleccionando las frases que contienen “Gy”. Luego se vuelven a unir y se presenta todo utilizando la función datatable() de DT con las opciones searchHighlight=TRUE y search=”GY”, para que aparezca GY en amarillo y poder localizar rápidamente los valores de dosis.

5.2.10. WordCloud utilizando la Taxonomia.

En este apartado se muestra una nube de palabras utilizando una taxonomía de toxicidades y tipos de tratamiento. Las toxicidades se obtienen del artículo de Emami B. “Tolerance of Normal Tissue to Therapeutic Radiation”[25] y se utiliza la función tdm_for_lsa() del paquete *pubmed.mineR* para calcular la matrix término-documento y luego la función wordcloud() del paquete wordcloud para dibujar la nube de términos.

5.2.11. WordCloud utilizando las palabras más frecuentes

Para generar la nube de palabras partiendo de los términos más frecuentes, primero se genera el Corpus con la función Corpus() de tm, de ese corpus se eliminan los número, la puntuación, se escribe todo en minúsculas, se eliminan las palabras comunes del idioma utilizado, y las terminaciones, manteniendo las raíces de las palabras. Se utiliza la función DocumentTermMatrix() para crear la matriz término documento y la función wordcloud() del paquete *wordcloud* para dibujar la nube de términos.

5.2.12. WordCloud utilizando MeSH

Para generar la nube de datos partiendo de los téminos MeSH, (acrónimo de Medical Subject Headings), se usa la función biblioAnalysis() y luego summary.bibliometrix(), para obtener la lista de términos MeSH y sus frecuencias y a función wordcloud() del paquete *wordcloud* para dibujar la nube de términos.

5.2.13. Similitudes por el coseno

A partir de la matriz término-documeno calculada con la función tdm_for_lsa() del paquete *pubmed.mineR* se calcula la similitud del coseno con la función cos_sim_calc() y se utiliza el paquete *igraph* para expresar los resultados de forma gráfica.

5.2.14. Similitudes por el coseno agrupando en comunidades

Con la función cluster walktrap(), que trata de encontrar subgrafos densamente conectados, también llamados comunidades en un gráfico a través de caminatas aleatorias. La idea es que los paseos aleatorios cortos tienden a permanecer en la misma comunidad y se utiliza el paquete *igraph* para expresar los resultados de forma gráfica.

5.2.15. Dendograma

Se hace una agrupación en clusters jerárquica, utilizando la función hclust() del paquete stats y con rect.hclust() se dibujan rectángulos alrededor de los clusters formados, para lo que hay que elegir en el menú de la izquierda “Total de clusters”. El método de unión usado es “ward.D”.

3.1.16. Cluster en grupos

Se muestran los mismos grupos que el dendograma, utilizando primero la función cutree(), para separar el arbol obtenido con hclust() en los grupos que se le

indique y fviz_cluster() del paquete factoextra para presentar los aglomerados en forma gráfica.

3.1.17. Tabla de clusters

Muestra una tabla indicando el número de componentes y los cinco términos más frecuentes en cada uno de los grupos anteriores.

3.1.18. Artículos de clusters

Se muestra de lista de los artículos pertenecientes a un grupo. Para elegir el grupo seleccionamos el “Seleccione cluster”

5.3. Paquete shiny

Shiny es un paquete de R que permite construir aplicaciones web interactivas a partir de los scripts de R utilizando la arquitectura Cliente-Servidor, esta librería genera el HTML5/JavaScript/CSS necesario para construir las Aplicaciones Web con parámetros y variables dinámicas que permiten interactuar con los datos sin necesidad de conocimientos de programación.

Las Aplicaciones Web desarrolladas con el paquete Shiny funcionan de la misma forma que otras aplicaciones Web, con la ventaja de que Shiny genera el código HTML necesario, permitiendo al usuario la manipulación de sus datos sin necesidad de manipular el código. Estas aplicaciones, están compuestas por un archivo **app.R** que aglutina todo el código, o por dos archivos **ui.R** y **server.R**, que separan la parte cliente de la parte servidor.

- **app.R:** fichero que contiene tanto los elementos de la interfaz como del servidor.
- **ui.R:** fichero donde se especifica la interfaz y la ubicación de los elementos en la pantalla, es decir, contiene la secuencia de comandos que controla el diseño y aspecto de la aplicación, es decir, recibe los inputs y muestra los outputs.
- **server.R:** fichero donde se encuentran las instrucciones que se ejecutan cada vez que el usuario hace un cambio o una interacción en pantalla, es decir, realiza los cálculos necesarios.

Shiny se basa en la programación reactiva, que vincula los valores de entrada con los de salida. Además de generar el HTML/JavaScript, ofrece widgets preconstruidos, facilitando la elaboración de aplicaciones web interactivas. En la Tabla 3, muestran los widgets de shiny:

Widget	Función	Argumentos comunes
Botón de acción	actionButton	inputId, label
casilla	checkboxInput	inputId, label, value
grupo de casillas	checkboxGroupInput	inputId, label, choices, selected
selección de fechas	dateInput	inputId, label, value, min, max, format
selección rango fechas	dateRangeInput	inputId, label, start, end, min, max, format
subir archivo	fileInput	inputId, label, multiple
campo numerico	numericInput	inputId, label, value, min, max, step
botón de selección	radioButtons	inputId, label, choices, selected

Tabla 3. *Widgets*

Los inputs que se introducen en ui.R, se envían a server.R y se utilizan para obtener los outputs. Las operaciones realizadas con los inputs en server.R, que dan como resultado los outputs, utilizan funciones renderTipo(), que se detallan a continuación en la Tabla 4:

Objeto que se obtiene	Uso
renderImage	Imagenes
renderPlot	Plots
renderPrint	Any printed output
renderTable	Dataframe, tabla
renderText	Texto
renderUI	Objeto Shiny tag o HTML

Tabla 4. *Funciones renderTipo()*

Los resultados Output() pueden ser de los tipos que se muestran en la Tabla 5:

Output	Inserta
dataTableOutput	una tabla interactiva
tableOutput	una tabla
imageOutput	imagen
plotOutput	un plot
htmlOutput	raw HTML
textOutput	texto
verbatimtextOutput	texto

Tabla 5. *Funciones Output()*

5.4. Diseño de la plataforma

Como se muestra en la Figura 5, se ha realizado un diseño sencillo para que su uso sea intuitivo, con pestañas en la parte superior, y una barra lateral a la izquierda usando las funciones tabsetPanel() y tabPanel().

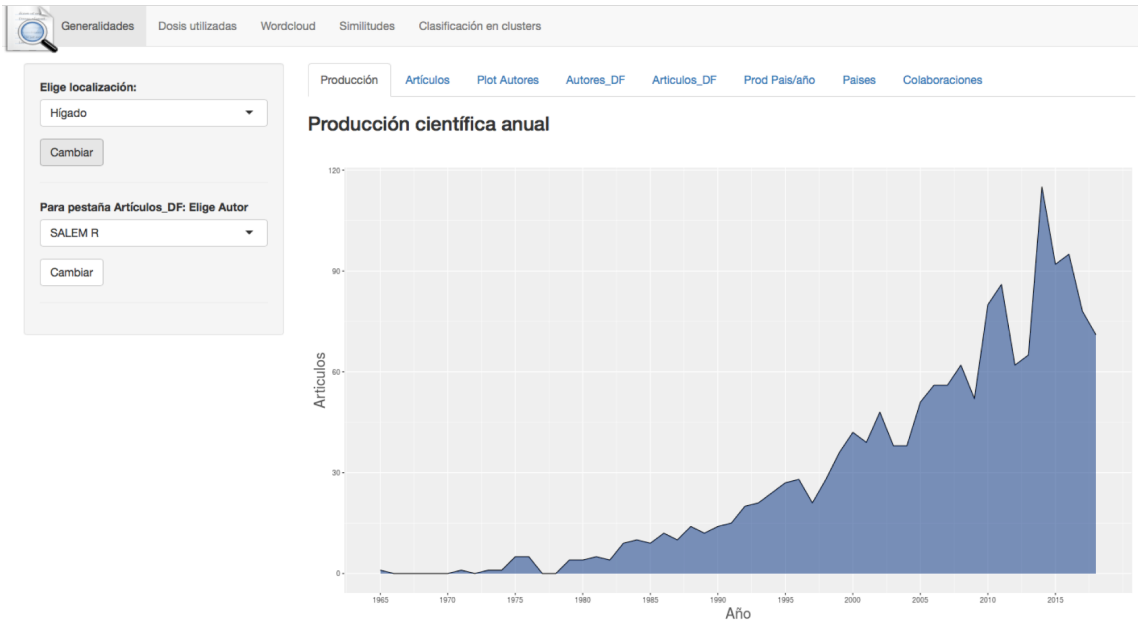

Figura 5. *Panel inicial*

En las pestañas de la parte superior se disponen de las siguientes opciones:

- **Generalidades:** Se extrae información de los metadatos, y se muestra en forma gráfica las publicaciones por año, autores y países más prolíficos, así como, colaboraciones entre los mismos.

- **Dosis utilizadas:** En este apartado se criba información sobre las dosis aplicadas, y se muestran las frases de los resúmenes que aportan información sobre la dosis administrada en distintas publicaciones.

- **Wordcloud:** Se genera una nube de términos frecuentes utilizando tres técnicas. distintas: una taxonomía, las palabras que aparecen con más frecuencia en el texto de los resúmenes y los MeSH (Medical Subject Headings).

- **Similitudes:** Se estudia la relación entre toxicidades y tipos de tratamiento utilizando LSA y se muestra en forma gráfica.

- **Clasificación en clusters:** En este apartado se separan los resúmenes en grupos y se muestra una lista de los téminos que definen cada grupo y los artículos que los forman..

La forma en la cual se relacionan las interacciones o cambios de los usuarios con los resultados, se realizan en el servidor mediante las variables input y output. En la interfaz se han creado distintas variables input en la llamada a selectInput() en las que se pueden elegir distintas localizaciones. Por localizaciones, entendemos las distintas zonas en las que se trata un tumor con radioterapia. Entre las distintas localizaciones se encuentran:

Hígado: fichero obtenido al buscar en PubMed "Liver toxicity radiotherapy".

Esófago: fichero obtenido al buscar en PubMed "Esophagus toxicity radiotherapy".

Próstata: fichero obtenido al buscar en PubMed "Prostate toxicity radiotherapy".

Mama: fichero obtenido al buscar en PubMed "Breast toxicity radiotherapy".

Cabeza-Cuello: fichero obtenido *al buscar* en PubMed "Head and neck toxicity radiotherapy".

Recto: fichero obtenido *al buscar* en PubMed "Rectal toxicity radiotherapy".

Pulmón: fichero obtenido al buscar en PubMed "Lung toxicity radiotherapy".

6. Resultados

Con esta aplicación se puede estudiar la relación entre toxicidades y tratamientos para distintas localizaciones y se va a aplicar a la radioterapia de próstata.

6.1. Aplicación al tema “toxicidades en tratamientos de próstata para distintos tipos de tratamiento”.

A continuación, se exponen los resultados obtenidos en cada uno de los apartados de esta aplicación web tras elegir la localización *Próstata* en el menú de la izquierda.

6.1.1. Producción científica anual.

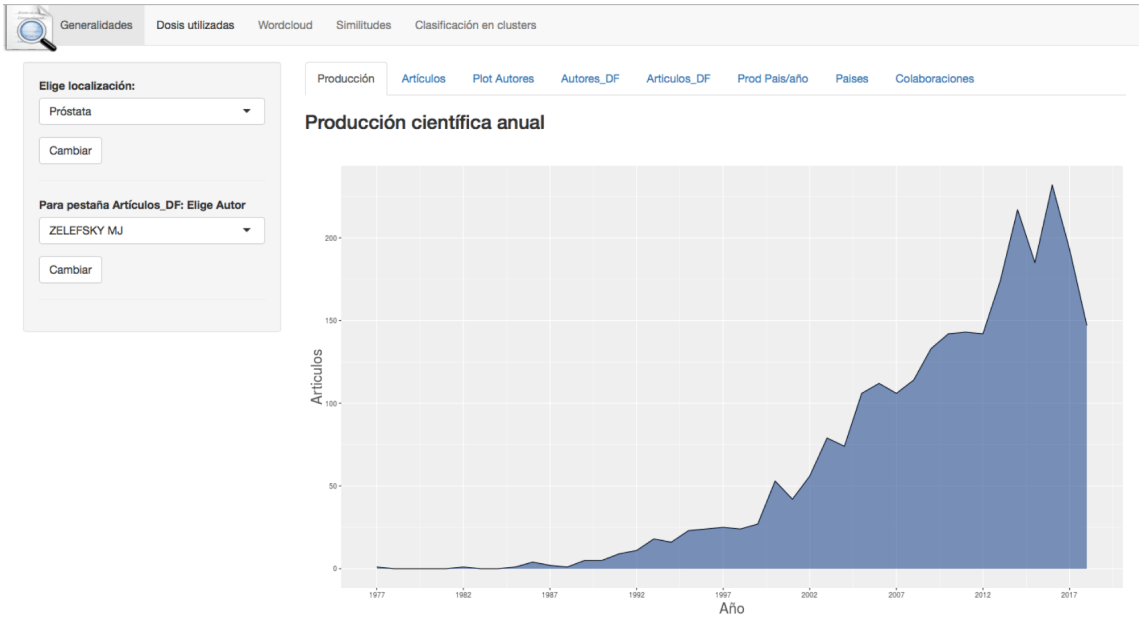

Figura 6. *Producción científica anual.*

En la Figura 6 se muestra la producción científica anual, es decir, el número de artículos publicados por año. Se puede observar que no es un tema nuevo, ya que se escribe sobre él desde hace más de 30 años. El número de artículos escritos sobre este tema ha aumentado mucho en los últimos años, estando por encima de 150 articulos por año en los últimos 4.

6.1.2. Lista de Artículos.

Eligiendo esta pestaña se obtiene una tabla con la lista de artículos que hablan sobre el tema elegido, es decir, toxicidades en radioterapia de próstata.

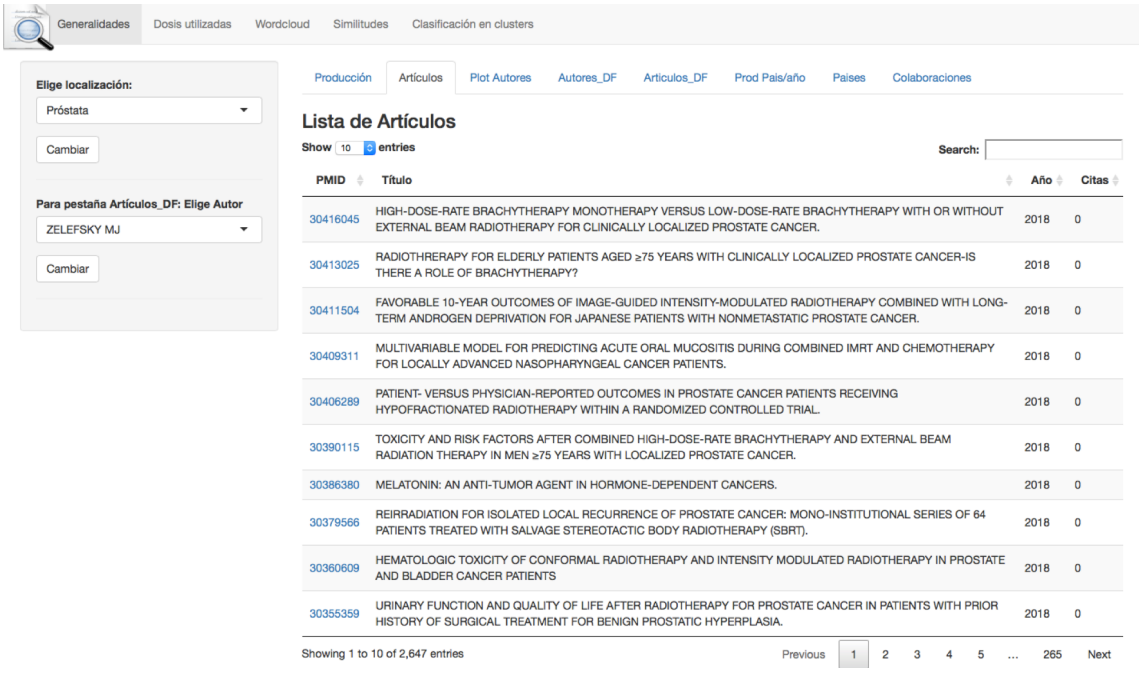

***Figura 7**. Lista de Artículos*

En la primera columna, se muestra el código PMID, al seleccionarlo se abre una nueva pestaña en PubMed donde se puede leer el texto completo del resumen. En la segunda columna, aparece el título y en la tercera y cuarta, año de publicación y el número de veces que se ha citado dicho artículo. Seleccionando las flechas situadas junto a “año” o “citas” se puede ordenar la tabla en modo cronológico, o por orden de influencia, entendiendo que los artículos más influyentes son los que más citas acumulan.

6.1.3. Autores más prolíficos.

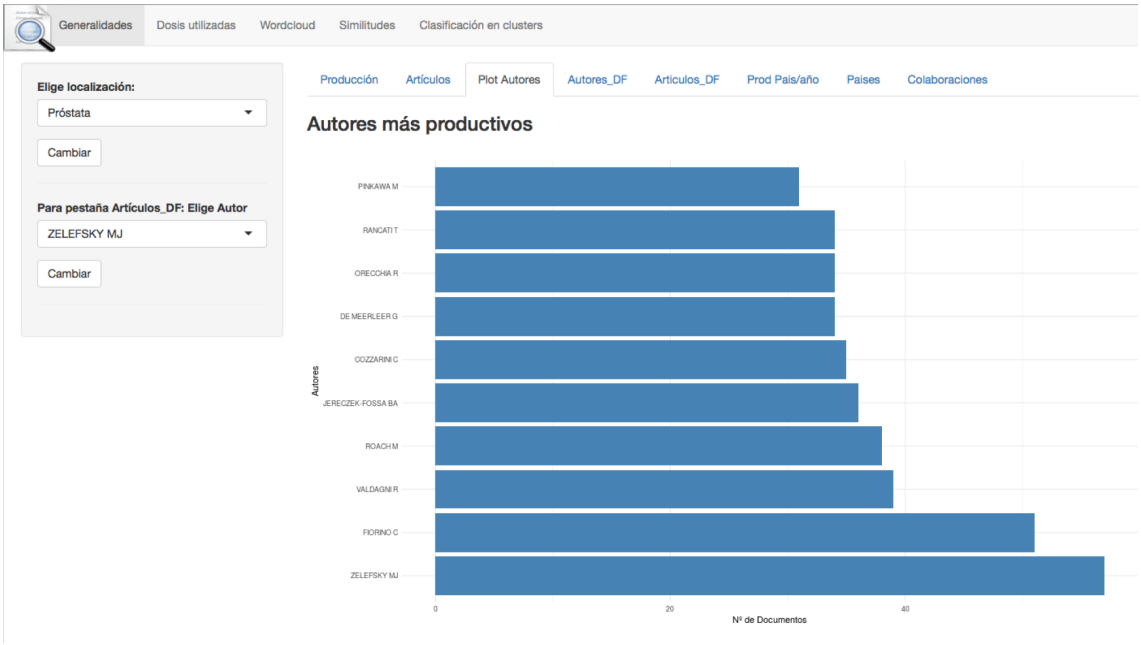

Figura 8. *Autores más prolíficos*

En el gráfico de barras de la Figura 8 se muestran los diez autores más prolíficos, siendo los dos primeros Zelefsky y Fiorino con un total de más de 50 artículos publicados.

6.1.4. Factor de Dominancia.

En este apartado, se calcula el factor de dominancia definido como el número de articulos en los que el autor, es autor principal entre el número de artículos en los que es coautor. En la tabla de la Figura 9 aparecen junto a la lista de los 10 autores más prolíficos, el factor de dominancia en la segunda columna. En la cuarta columna aparencen el número de artículos publicados como autor principal, y en la tercera el número de total artículos publicados como coautor. En las últimas dos columnas se presenta el ranking por artículos publicados y por factor de dominancia.

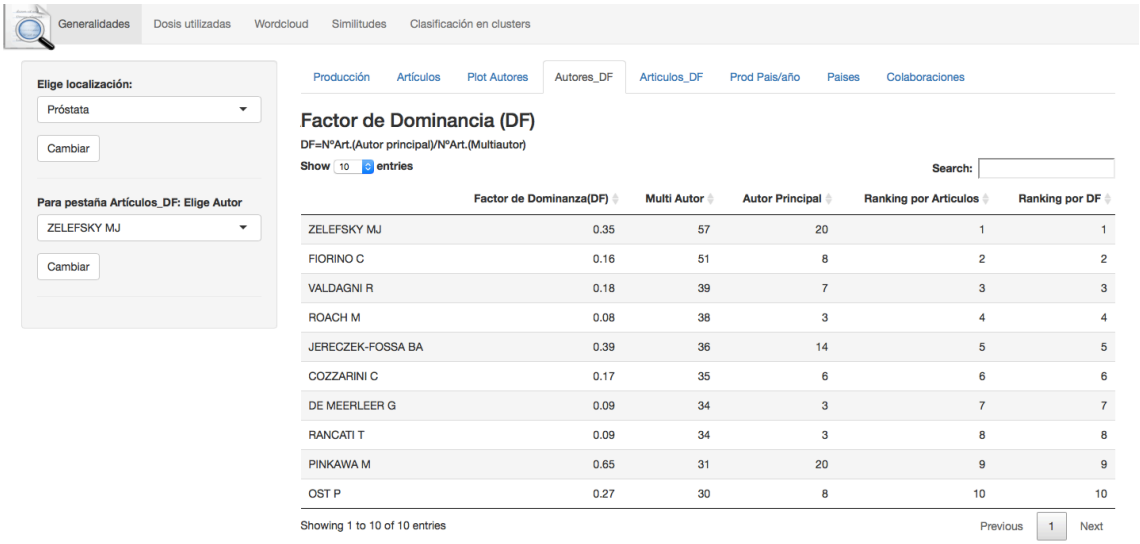

Figura 9. *Factor de Dominancia*

Se observa que, aunque Zelefsky MJ, es el autor más prolífico teniendo en cuenta el número de publicaciones totales, Pinkawa M ha publicado más veces como autor principal en relación a las que ha sido coautor, por lo que su factor de dominancia es mayor.

6.1.5. Artículos de los autores más prolíficos.

Al elegir en el menú de la izquierda el nombre de un autor entre los 10 más prolíficos, se muestra la tabla con los artículos escritos por dicho autor. Ésta es otra forma de ordenar artículos según su influencia, ya que se considera que los autores que más publicaciones acumulan sobre un tema escribirán también artículos que acumulen muchas citas. Si se identificara la influencia de una publicación únicamente por las citas acumuladas, se minusvaloraría las publicaciones recientes por la imposibilidad de ser citadas. Como se muestra en la Figura 10 uno de los artículos de Zelefsky MJ (el autor más prolífico), acumula más de 112 citas.

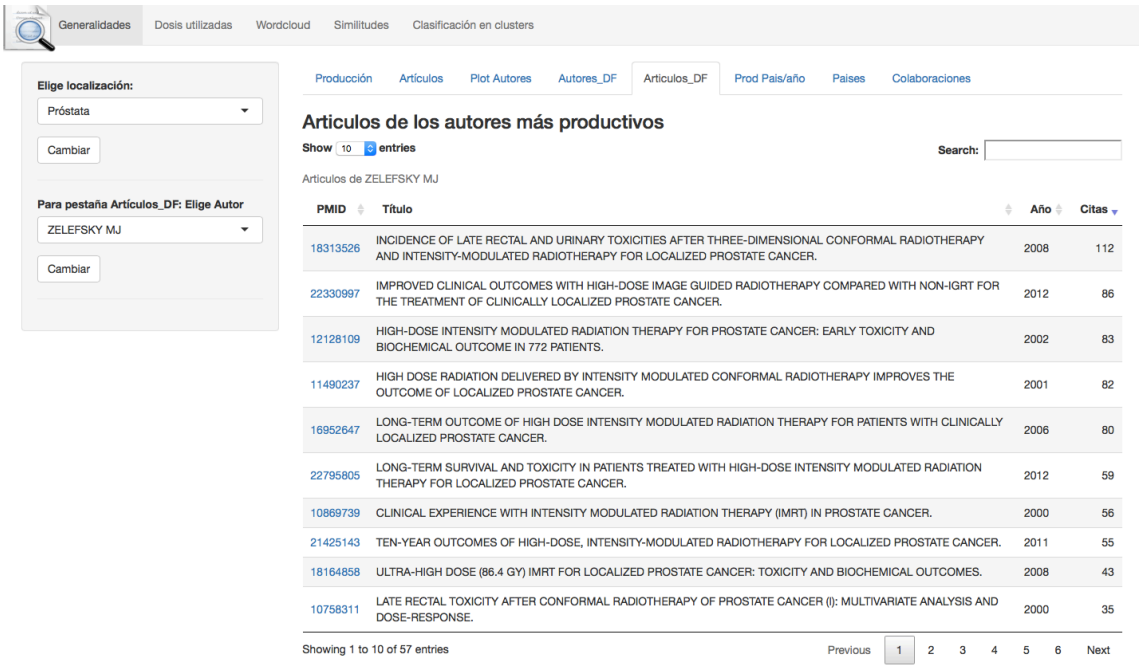

Figura 10. *Artículos de los autores más prolíficos*

6.1.6. Publicaciones de paises por año.

En la Figura 11, se muestra un gráfico tipo h*eatmap*, donde se visualiza la evolución del número de publicaciones sobre el tema de estudio para distintos países. Se observa que el país que más ha publicado a lo largo de los últimos año es Estados Unidos, llegando al máximo de publicaciones en el 2016. Otros paises que tienen muchas publicaciones sobre toxicidades en tratamiento de radioterapia de próstata son Inglaterra, Alemania e Irlanda.

Figura 11. *Publicaciones de países por año*

6.1.7. Paises más productivos.

En el gráfico de barras de la Figura 12, se muestran el número total de artículos publicados por país sobre toxicidades en tratamientos de radioterapia de próstata. La parte roja de la barra se refiere a las publicaciones del país en colaboración con otros países y en azul los artículos escritos en solitario. Se observa que Estados Unidos es el país más productivo en solitario, seguido de Italia y Canadá, mientras que Australia es el más productivos en colaboraciones internacionales seguido de Italia.

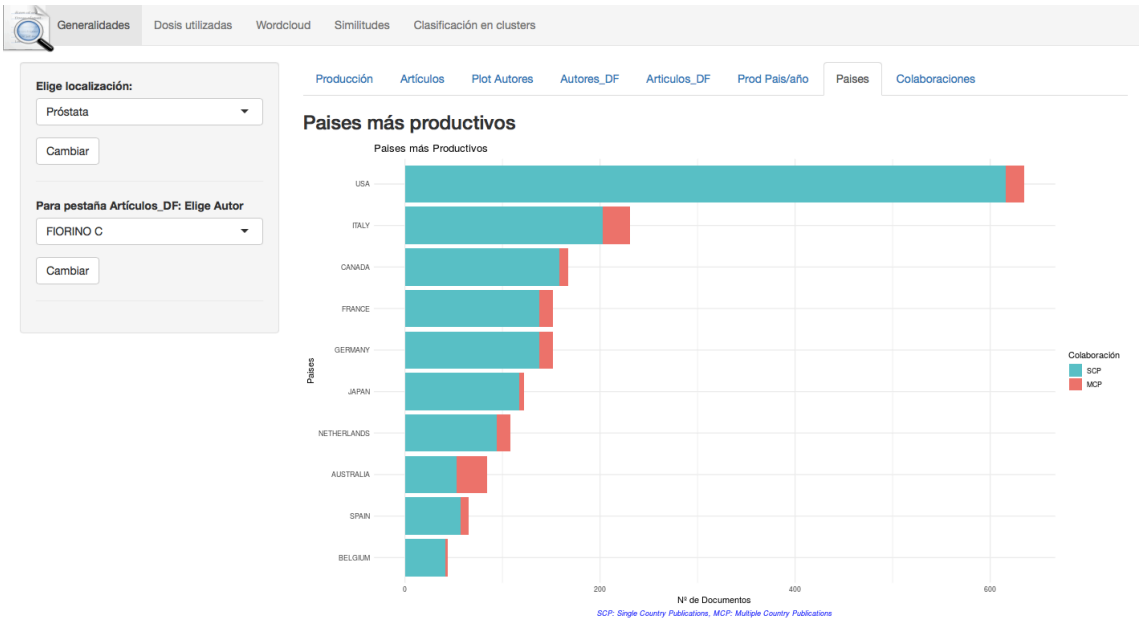

***Figura 12**. Países más productivos*

6.1.8. Colaboraciones entre paises.

El gráfico circular de la Figura 13, es muy interesante ya que en él se pueden observar las colaboraciones entre los distintos países. Se aprecia que Estados Unidos tiene muchas relaciones de colaboración con otros países, seguido por Alemania, que colabora sobretodo con países europeos.

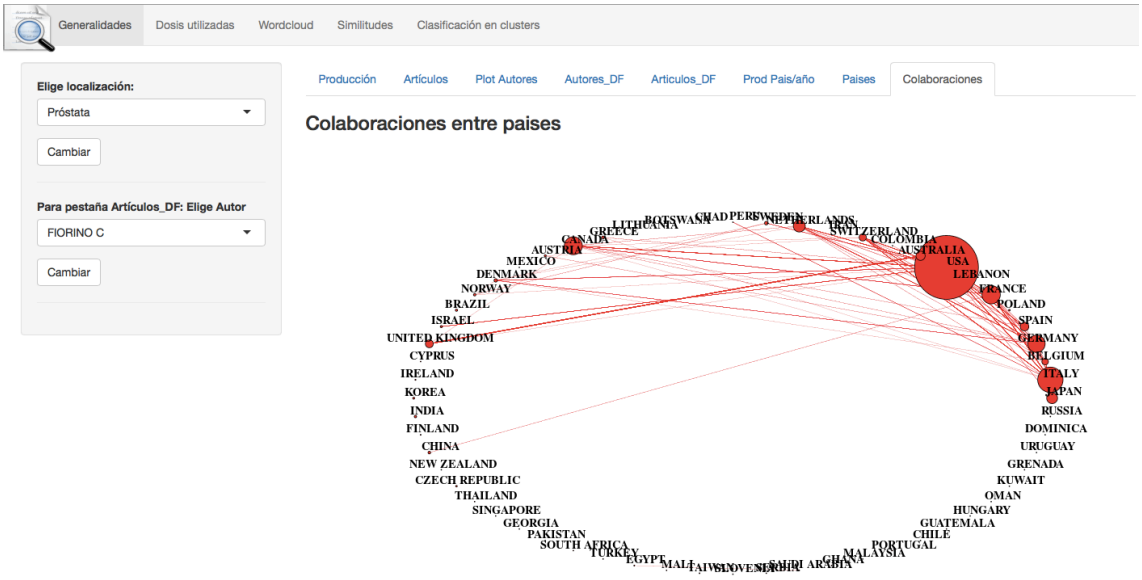

Figura 13. *Colaboraciones entre países.*

6.1.9. Dosis utilizadas.

En el estudio de toxicidades en tratamientos de radioterapia de próstata, así como, para otras localizaciones, es importante saber la dosis de radiación administrada en los distintos estudios, ya que, los efectos adversos del tratatamiento, es decir, las toxicidades, dependen tanto de la dosis total como de la dosis por fracción.

En este apartado, se elige localización (*Próstata*) y año, y en la tabla se visualizan las frases en las que se habla de dosis y se muestra en amarillo “Gy” que es la unidad de dosis. Así, en la tabla de la Figura 14, se pueden identificar rápidamente distintos esquemas de dosis y fraccionamiento, como, por ejemplo, 78Gy (39 fracciones de 2 Gy), que es el más usado, o 36.25 Gy (5 fracciones de 7.25Gy), lo que corresponde a un tratamiento hipofraccionado. Al disponer de la información en forma de tabla, resulta sencillo seleccionar el artículo con el esquema de dosis y fraccionamiento que resulte más interesante y leer el resumen completo.

Es interesante hacer la búsqueda por años, ya que los fraccionamientos evolucionan con el transcurso de los años por la innovación tecnológica. Siendo la tendencia general la aplicación de dosis más elevadas, aumentando la dosis por fracción y/o disminuyendo el número de fracciones. Se puede ver también las veces que se ha citado un artículo así como su año de publicación para hacerse una idea de la influencia del mismo, y leer el resumen completo seleccionando su código PMID.

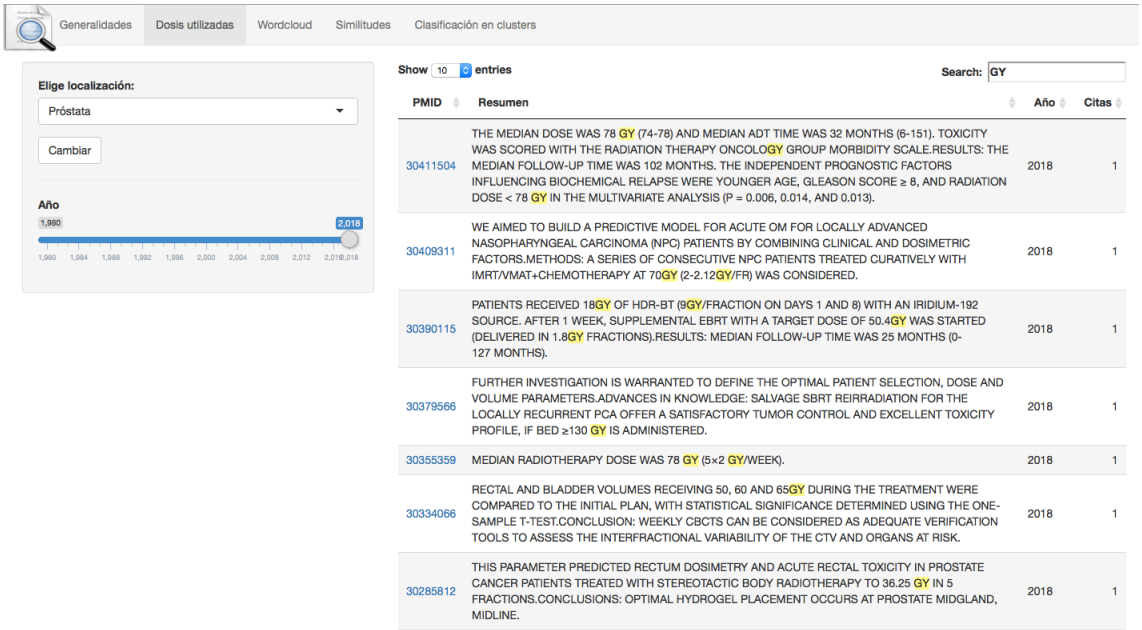

Figura 14. *Dosis Utilizadas*

6.1.10. WordCloud utilizando la Taxonomia.

En este apartado se muestra una nube de términos, utilizando una taxonomía de toxicidades y tipos de tratamiento. Las toxicidades se obtienen del artículo de Emami B. “Tolerance of Normal Tissue to Therapeutic Radiation”[23]

La ***Wordcloud*** muestra la evolución de términos frecuentes en los textos de las sinopsis con tres métodos distintos (taxonomía, palabras frecuentes y MeSH). De los tres métodos, el uso de una taxonomía es el que mejor funciona para mostrar dicha evolución. Al seleccionar este método y distintos intervalos de tiempo, se observa como los tratamientos de próstata evolucionan de 3D (Conformal, Conformal Radiotherapy, CRT,) a IMRT y a SBRT, y el interés creciente de los tratamientos de braquiterapia de próstata con semillas hasta aproximadamente el año 2000 decreciendo luego.

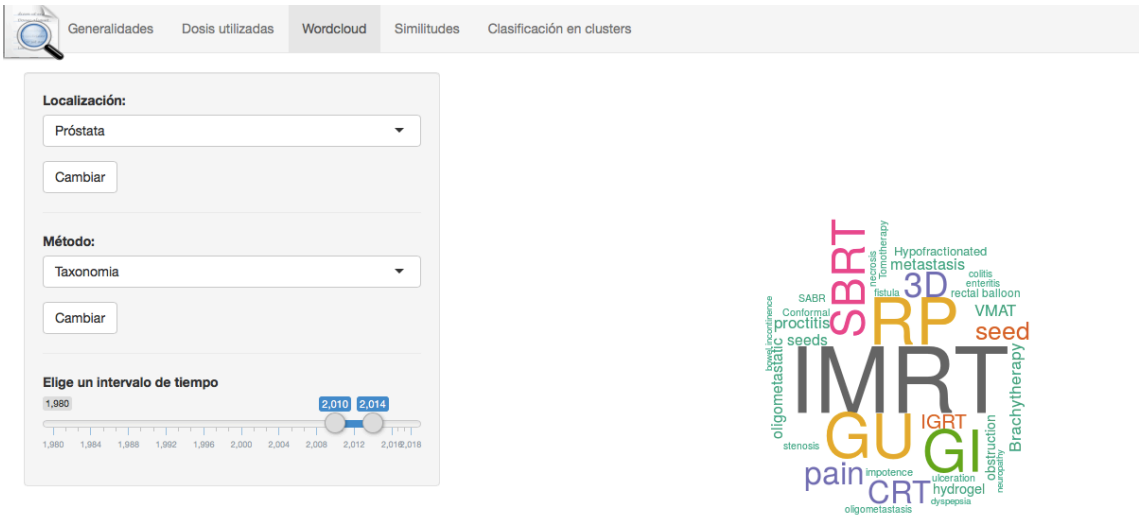

Figura 15. *Wordcloud con Taxonomía*

6.1.11. WordCloud utilizando las palabras más frecuentes

Utilizando este método para elaborar la wordclouds, se observa que los términos con máxima ocurrencia a lo largo de los años son *patient, toxic, treatment* y *dose*.

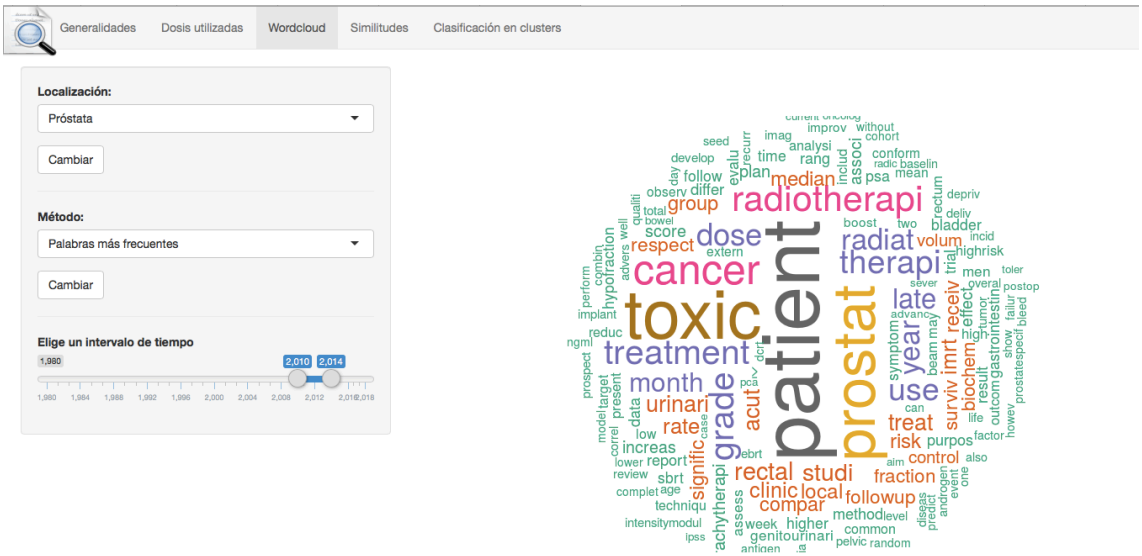

***Figura 16**. WordCloud con palabras frecuentes*

6.1.12. WordCloud utilizando MeSH

El témino MeSH, (acrónimo de Medical Subject Headings), se refiere a un vocabulario terminológico controlado para publicaciones de artículos y libros de ciencia, creado por la Librería Nacional de Medicina (NLM).

Como se observa en la Figura 17 los téminos mas frecuentes son *radiotherapy, methods, humans,* y *adverse effects*.

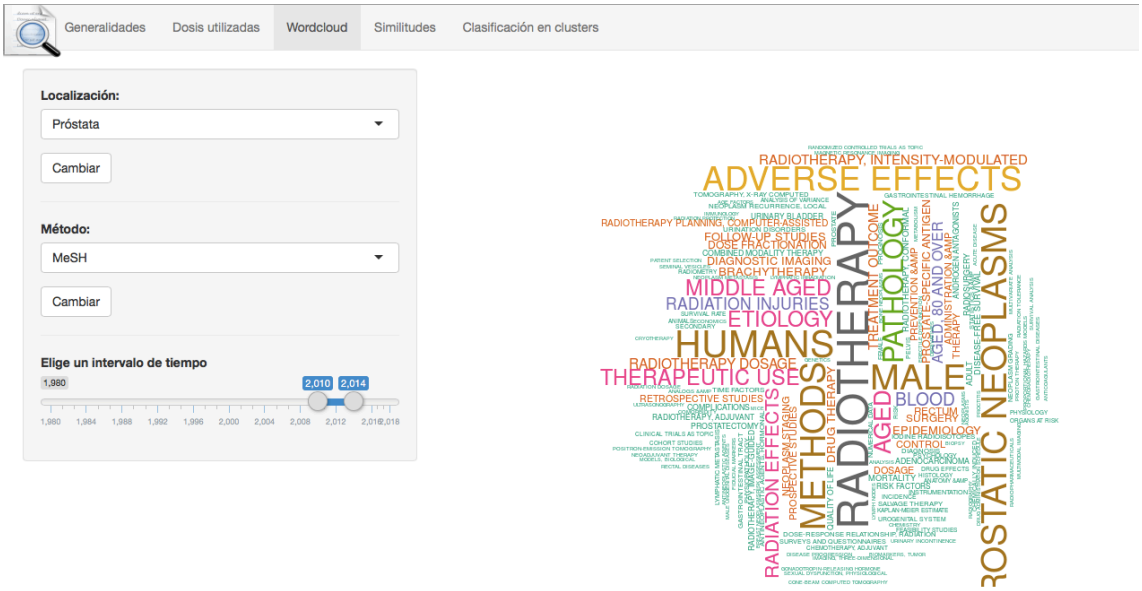

Figura 17. *WordCloud utilizando MeSH*

6.1.13. Similitudes por el coseno

En la Figura 18 se observan las similitudes entre distintos términos. Aumentando el nivel de correspondencia aparecen menos relaciones entre términos. Con nivel de correspondencia 0.3 y año 2017:

- Se observan relaciones entre *brachytherapy*, *seed* y *seeds*, ya que las semillas son tratamientos de braquiterapia.
- El término *seeds* está relacionado con *proctitis*. La **proctitis** es un proceso inflamatorio propio del recto, que afecta fundamentalmente a la mucosa, y que puede producirse por distintas causas, en este caso por la migración de alguna de las semillas de la próstata al recto.
- Los términos *SBRT* y *SABR*, están relacionados con *oligometastatic*, ya que ambos tratamientos se usan para las oligometástasis.
- El término 3D se relacionado con Conformal, conformal radiotherapy, y CRT porque se refieren al mismo tipo de tratamiento.
- *Fístula*, *proctitis* y *ulceration* son términos que están relacionados ya que la proctitis puede generar fístulas y ulceraciones.

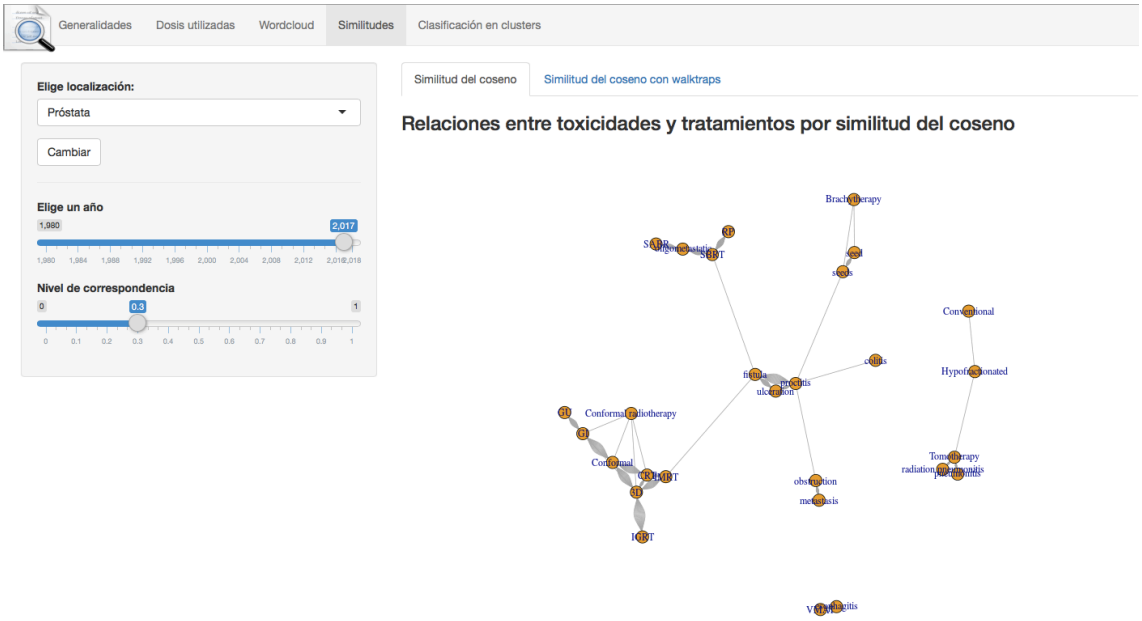

***Figura 18**. Relaciones entre toxicidades y tratamientos por similitud del coseno*

6.1.14. Similitudes por el coseno agrupando en comunidades

Los subgrafos densamente conectados, también llamados ***comunidades*** se muestran en la Figura 19. Eligiendo el mismo nivel de correspondencia y año que en el apartado anterior (0.3 y 2017) se observan las mismas relaciones encontradas anteriormente agrupadas en comunidades:

- Los términos *brachytherapy*, *seed* y *seeds*, se agrupan en una comunidad color verde.
- El término *seeds* está relacionado con *proctitis* y forma junto a *fistula*, *ulceration* y *colitis*. una comunidad mostrada en color lila.
- Los términos *SBRT* y *SABR*, que están relacionados con *oligometastatic*, forman otra comunidad mostrada en color verde claro junto a *RP*.
- Los términos *3D, Conformal, conformal radiotherapy*, y *CRT* relacionados entre si por referirse al mismo tipo de tratamiento, forman una comunidad mostrada en color azul con otros términos como *IGRT, IMRT, GU y GI*.
- La comunidad color salmón es la formada por *metastasis y obstruction* que está relacionada con *proctitis*.

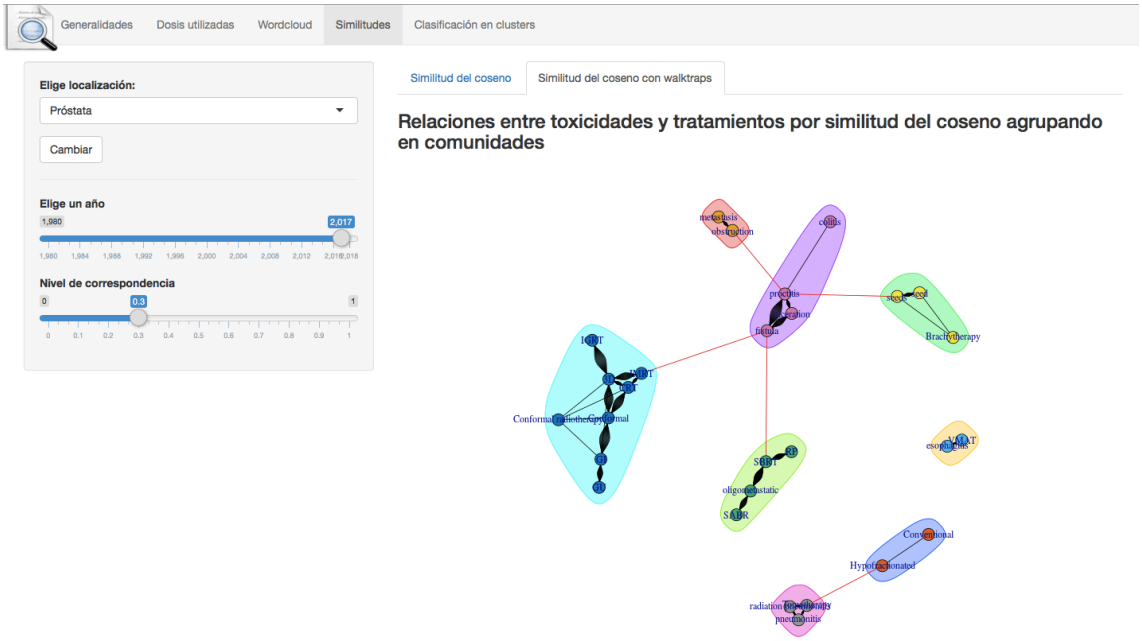

Figura 19. *Similitudes por el coseno agrupando en comunidades*

6.1.15. Dendograma

Como se muestra en la Figura 20, en este apartado se hace una agrupación jerárquica de los resúmenes. Se elige el año de publicación (para que el número de abstracts no sea muy grande) y el número de grupos a formar. Los resúmenes que pertenecen al mismo grupo están recuadrados en color rojo.

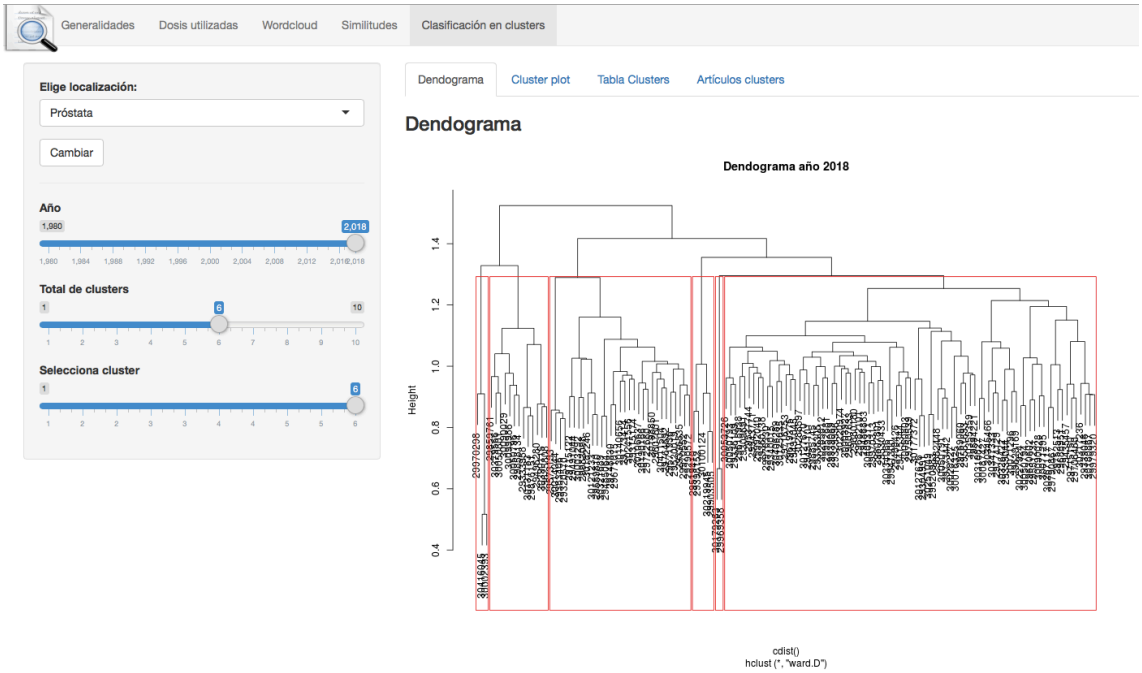

***Figura 20**. Dendograma*

6.1.16. Abstracts en grupos

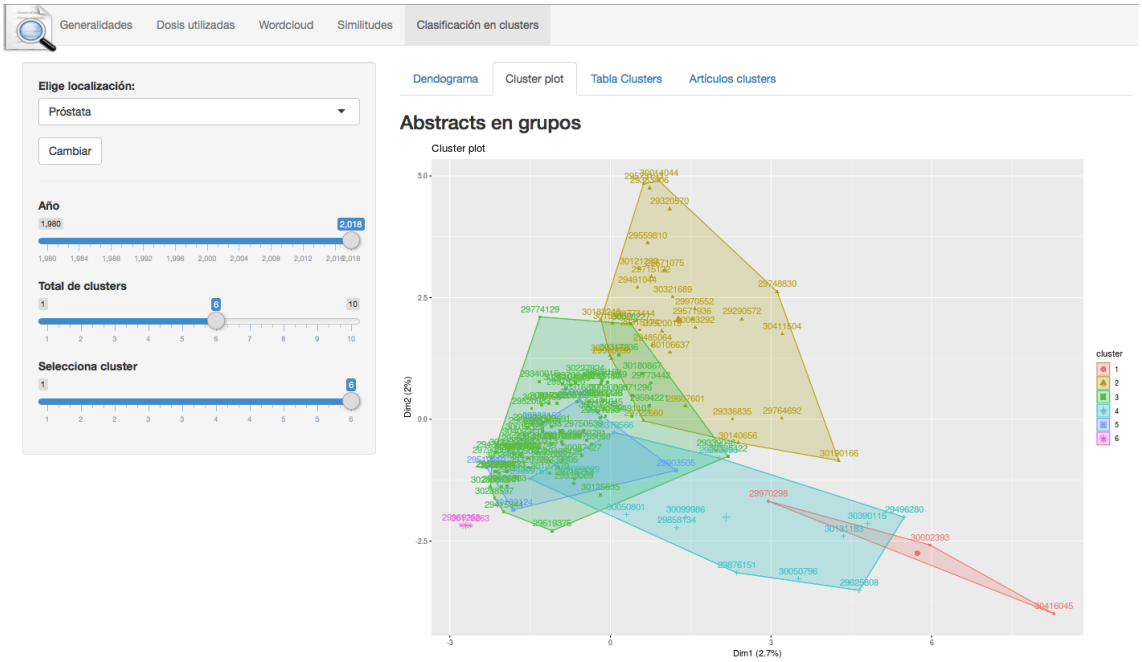

Figura 21. *Abstracts en grupos*

En la Figura 21, se muestran los mismos grupos que en el apartado anterior, pero representados en dos ejes. Se presentan los grupos con distintos colores: El grupo 1 en rojo, el 2 en amarillo, el 3 en verde el 4 en turquesa, el 5 en azul y el 6 en rosa.

6.1.17. Tabla de clusters

En la tabla de la Figura 22, se muestran los cinco términos más frecuentes en cada uno de los grupos anteriores y su tamaño. Eligiendo el año 2018 y 6 clusters y analizando los téminos que definen los grupos se puede entender el motivo de la agrupación. Por ejemplo, el cuarto grupo consta de 13 resúmenes que hablan sobre tratamientos de braquiterapia y en el sexto de dos artículos de *radiomics*, que es un tema novedoso en radioterapia.

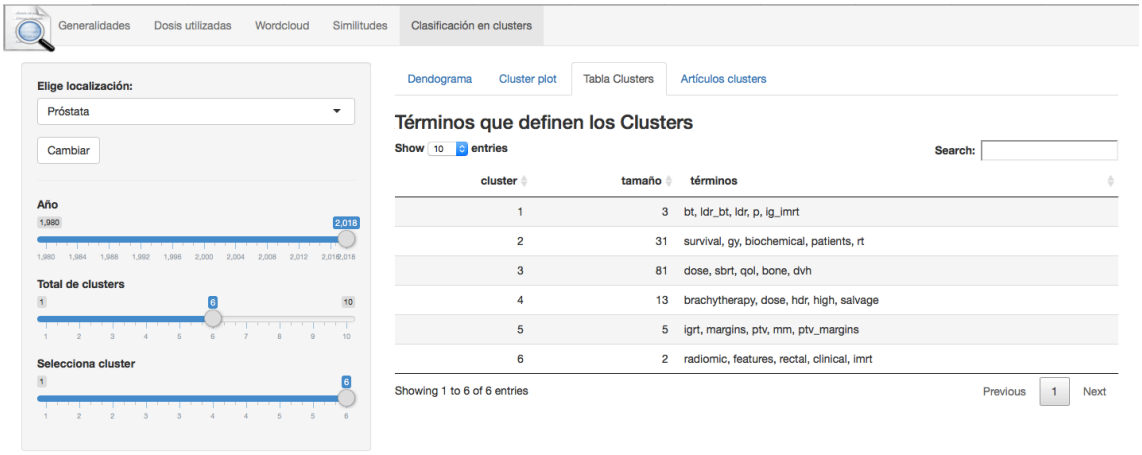

***Figura 22**. Tabla de clusters.*

6.1.18. Artículos de clusters

En la tabla de la Figura 23, se muestra de lista de los artículos pertenecientes a un grupo. Para elegir el grupo en “Seleccionar cluster” se elige el 6 y se obtiene la lista de todos los artículos que componen el grupo 6. Son dos artículos sobre *radiomics*, que es un campo de estudio médico que pretende extraer una gran cantidad de características cuantitativas de imágenes médicas mediante algoritmos de caracterización de datos. En este caso los dos artículos analizan imágenes de resonancia magnética para analizar cambios en la pared rectal y predecir toxicidad en el recto.

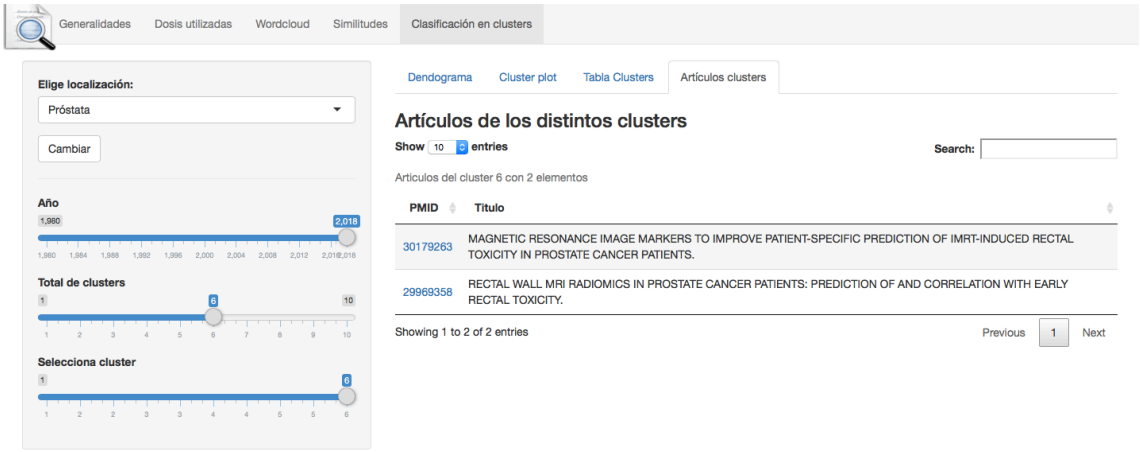

***Figura 23**. Artículos de uno de los clusters*

7. Conclusiones

La aplicación web, permite tabular la información obtenida tras una búsqueda en PubMed y presentarla en forma gráfica, lo que agiliza su análisis, al ser realizado con mayor rapidez y eficacia. Su estructura en bloques la hace de sencillo manejo, y la aplicación de métodos de minería de texto y estadística multivariante, permite la extracción de información útil e importante de los datos no estructurados de las sinopsis.

La aplicación permite evaluar la información desde diferentes puntos de vista : Generalidades, Dosis utilizadas, Worcloud, Similitudes y Clasificación en Cluster.

En primer lugar, en ***Generalidades***, se ofrece una visión general, en forma gráfica, del estado del arte del tema de la búsqueda, de la evolución en el interés que suscita (creciente o decreciente) y su vigencia (clásico o reciente), así como, el interés que suscita por países y la colaboración entre los mismos. Además, ordena los artículos en forma de tabla facilitando la selección de los más influyentes, entendiendo como tales, los más citados, o los escritos por autores más prolíficos.

En el bloque ***Dosis utilizadas***, aplicándolo al estudio de la relación entre tratamientos y toxicidades en radioterapia de próstata, se permite identificar rápidamente distintos esquemas de dosis y fraccionamiento, como, por ejemplo, el más usado, 78Gy (39 fracciones de 2 Gy), o 36.25 Gy (5 fracciones de 7.25Gy), lo que corresponde a un tratamiento hipofraccionado. El disponer de la información en forma de tabla, permite identificar los artículos con el esquema de dosis y fraccionamiento deseado, sin necesidad de leer el resumen completo.

El ***Wordcloud*** muestra la evolución de términos frecuentes en los textos de las sinopsis con tres métodos distintos (taxonomía, palabras frecuentes y MeSH). De los tres métodos, el uso de una taxonomía es el que mejor funciona para mostrar dicha evolución. Al seleccionar este método y distintos intervalos de tiempo, se observa como los tratamientos de próstata evolucionan de 3D (Conformal, Conformal Radiotherapy, CRT,) a IMRT y a SBRT, así como el descenso en el interés por los tratamientos de braquiterapia de próstata con semillas, desde su punto álgido en el año 2000.

En el bloque ***Similitudes***, se aplican técnicas de minería de texto para buscar similitudes entre distintos téminos, encontrando relaciones entre SBRT y metástasis. Se aprecian también similitudes entre “braquitherapy”, “seed” y “seeds”, ya que las semillas son tratamientos de braquiterapia, y “3D” con “Conformal”, “Conformal Radiotherapy”, y “CRT” al referirse al mismo tipo de tratamiento.

Por último, en el bloque ***Clasificación en Clusters***, se agrupan los resúmenes indicando los términos más frecuentes por grupo, mostrando una tabla con los artículos pertenecientes a cada grupo para su posterior análisis.

Respecto al seguimiento de la planificación, sería mejor disponer del tiempo conjunto de elaboración de la memoria y la presentación ya que se ambas tareas pueden realizase de forma paralela.

Una limitación de la aplicación, y por lo tanto una mejora potencial, es que la búsqueda está acotada al tema *toxicidades en tratamientos de radioterapia* y el usuario sólo puede manejar los archivos guardados en la aplicación. Se podría generalizar, añadiendo una

opción para realizar la búsqueda, formateo de la información, así como, su posterior lectura para proceder al análisis con las herramientas de la aplicación. Se necesitaría también facilitar la lectura de una taxonomía para los bloques ***Wordcloud*** y ***Similitudes***. Esta generalización estaba fuera de los objetivos de este trabajo por limitaciones de tiempo.

Las líneas de trabajo futuro serán, por una parte, explorar el uso de ontologías y por otra ampliar los métodos de clasificación con otras técnicas de análisis multivariante y distintos algoritmos de machine learning.

8. Glosario

GPL: General Public License
TFM: Trabajo Final de Master
NLM: National Library of Medicine
NCBI: National Center for Biotechnology Information
MeSH: Medical Subject Headings
PMID:PubMed Identification
LSA:Latent semantic analysis
SVD: Descomposición en valores singulares
SEFM: Sociedad Española de Física Médica
SEPR: Sociedad Española de Protección Radiológica
RION: Radiation-Induced Optic Neuropathy
SNHL: Sensorineural Hearing Loss
ORN: Osteoradionecrosis
RP: Radiation Pneumonitis
RIDL: Radiation-induced liver disease
GU: Gastrourinal
GI: Gastrointestinal
IMRT: Intensity Modulated Radiotherapy
SBRT: Stereotactic Body Radiotherapy
VMAT: Volumetric Modulated Arc Therapy
CRT: Conformal Radiotherapy
SABR: Stereotactic ablatie Radiotharapy
IGRT: Image guided radiotherapy

9. Bibliografía

1. http://www.ncbi.nlm.nih.gov/pubmed

2. Allahyari, Mehdi, Seyed Amin Pouriyeh, Mehdi Assefi, Saied Safaei, Elizabeth D. Trippe, Juan B. Gutierrez, y Krys J. Kochut. 2017. «**A Brief Survey of Text Mining: Classification, Clustering and Extraction Techniques**». *CoRR* abs/1707.02919.

3. Gaikwad, Sonali Vijay, Archana Chaugule, y Pramod Patil. 2014. «**Article: Text Mining Methods and Techniques**». *International Journal of Computer Applications* 85 (17): 42-45.

4. Talib, Ramzan, Muhammad Kashif Hanif, Shaeela Ayesha, y Fakeeha Fatima. 2016. «**Text Mining: Techniques, Applications and Issues**». *International Journal of Advanced Computer Science and Applications* 7 (11). The Science; Information Organization. doi:10.14569/IJACSA.2016.071153

.

5. Sharma S,Gupta V. **Recent Developments in Text Clustering Techniques.** *International Journal of Computer Aplications(0975-8887)* 37(6): 14-19

6. Welbers, Kasper, Wouter Van Atteveldt, y Kenneth Benoit. 2017. **Text Analysis in R**. *Communication Methods and Measures* 11 (4). Routledge: 245-65. doi:10.1080/19312458.2017.1387238.

7. Feinerer, Ingo, Kurt Hornik, y David Meyer. 2008. **Text Mining Infrastructure in R**. *Journal of Statistical Software, Articles* 25 (5): 1-54. doi:10.18637/jss.v025.i05.

8. Kenneth Benoit, Kohei Watanabe. 2018. **Quantitative Analysis of Textual Data.** http://quanteda.io.

9. Jurka T.P,Collingwood,L.2013.**RTextTools:A Supervised Learning Package for Text Classification.** The R Journal 5: 6-12.

10. Rani, J. Sharmila, Ab Rauf Shah, y Srinivasan Ramachandran. 2015. **pubmed.mineR: An R package with text-mining algorithms to analyse PubMed abstracts.** *Journal of Biosciences* 40: 671-82.

11. Kovalchik, Stephanie. 2017. «Download Content from NCBI Databases».https://cran.r-project.org/web/packages/RISmed/RISmed.pdf.

12. Aria, M. & Cuccurullo, C. (2017). *bibliometrix*: **An R-tool for comprehensive science mapping analysis**, *Journal of Informetrics*, 11(4), pp 959-975, Elsevier, DOI: 10.1016/j.joi.2017.08.007 (link)

13. Cohen KB, Roeder C, Xia J. 2016. **Reproducibility in Natural Language Processing: A Case Study of Two R Libraries for Mining PubMed/MEDLINE**, 6-12.

14. Korhonen, Anna, Ilona Silins, Lin Sun, y Ulla Stenius. 2009. **The first step in the development of text mining technology for cancer risk assessment: identifying and organizing scientific evidence in risk assessment literature**. *BMC Bioinformatics* 10: 303-3.

15. Sang-Jun Y. et al. **A dataminig approach to selecting herbs with similar efficacy: Targeted selection methods based on medical subject heading (MeSH).** *Journal of Ethnopharmacology* 182 (2016) 27-34

16. Smith P.F. **Review: On the Application of Multivariate Statistical and Data Mining Analyses to Data in Neuroscience**. *The Journal of Undergraduate Neuroscience* Education(JUNE), Spring 2018, 16(2)

17. Martinez-Garcia et al. **A systematic approach to analyze the social determinants of cardiovascular disease.** *PLOS ONE* January 25,2018 https://doi.org/10.1371/journal.pone.0190960

18. http://www.bibliometrix.org/biblioshiny.html

19. Zhang, Y., Sarkar, I. N., & Chen, E. S. (2014). **PubMedMiner: Mining and Visualizing MeSH-based Associations in PubMed**. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, *2014*, 1990-9.

20. Lin S.M., et al. **MedlineR: an open source library in R for Medline literature data mining.** Bioinformatics. 2004;20:3659–3661.

21. https://www.nlm.nih.gov/bsd/mms/medlineelements.html

22. Landauer, Th., y Dumais, S. (1997) **A solution to Platos problem: the latent semantic analysis theory of acquisition, induction, and representation of knowledge**. *Psychological Review*, 104, 211-240.

23. G. Salton, A. Wong, and C. S. Yang. 1975. **A vector space model for automatic indexing. Commun.** ACM 18, 11 (November 1975), 613-620. DOI=http://dx.doi.org/10.1145/361219.361220

24. https://github.com/massimoaria/bibliometrix/tree/master/R

25. Emami B. **Tolerance of normal tissue to therapeutic radiation.** *Reports of Radiotherapy and Oncology* 2013;1:35-48.

10. Anexos

Anexo 1. Código fuente de la aplicación web

```
library(shiny)

library(knitr)
# paquetes para mineria en PubMed
library(pubmed.mineR)
library(RISmed)
library(easyPubMed)
library(bibliometrix)
# paquetes para mineria no específicos de pubMed
library(tm)
library(textmineR)
library(wordcloud)
library(lsa)
#paquetes para clusters
library(proxy)
library(stats)
library(factoextra)
library(stopwords)
# paquetes para visualizar gráficos
library(ggplot2)
library(gplots)
library(igraph)
# paquetes para realizar tablas
#library(kableExtra)
library(DT)

#####################################################################

localizacion_df <<- list("Hígado"= "df_liver_tox_rad",
                         "Esófago" = "df_esoph_tox_rad",
                         "Próstata" = "df_prost_tox_rad",
                         "Mama" = "df_breast_tox_rad",
                         "Cabeza-Cuello" = "df_H&N_tox_rad",
                         "Recto"= "df_rect_tox_rad",
                         "Pulmón"= "df_lung_tox_rad")


localizacion <<- list("Hígado"= "liver_tox_rad",
                      "Esófago" = "esoph_tox_rad",
                      "Mama" = "breast_tox_rad",
                      "Próstata" = "prost_tox_rad",
                      "Cabeza-Cuello" = "H&N_tox_rad",
                      "Recto"= "rect_tox_rad",
                      "Pulmón"= "lung_tox_rad")

metodo <<- list("Taxonomia"="1",
                "Palabras más frecuentes"="2",
                "MeSH"="3")

#metodo_cl <<- list("ward.D"="ward.D",
#                   "ward.D2" = "ward.D2",
#                   "single" = "single",
#                   "complete" = "complete",
#                   "average" = "average",
#                   "mcquitty" = "mcquitty",
#                   "median" = "median",
#                   "centroid" = "centroid")
#####################################################################
#             UI
```

```
#########################################################################

ui = tagList(

 navbarPage(

  titlePanel(title =
           div(" -   - ",
              img(
                src = "lupa1.png",
                height = 65,
                width = 75,
                style="position:absolute;left:1px;margin:-40px 0px;z-index:1000000;"
                #style = "margin:0px 0px"
              )
           )
  ),
  tabPanel("Generalidades",
        sidebarPanel(
          selectInput(inputId= "selection_df",
                   label="Elige localización:",
                   choices = localizacion_df),
          actionButton("update", "Cambiar"),
          hr(),

          selectInput(inputId= "selection_AU",
                   label="Para pestaña Artículos_DF: Elige Autor",
                   choices = ""),
          actionButton("update", "Cambiar"),
          hr(),

          width = 3
        ),
        mainPanel(
          tabsetPanel(
            tabPanel("Producción",
                   h3("Producción científica anual"),
                   plotOutput("produccion")
            ),
            tabPanel("Artículos",
                   h3("Lista de Artículos"),
                   dataTableOutput("articulos")
            ),
            tabPanel("Plot Autores",
                   h3("Autores más prolíficos"),
                   plotOutput("autores")
            ),
            tabPanel("Autores_DF",
                   h3("Factor de Dominancia (DF)"),
                   h5("DF=NºArt.(Autor principal)/NºArt.(Multiautor)"),
                   dataTableOutput("dominancia")
            ),
            tabPanel("Articulos_DF",
                   h3("Articulos de los autores más productivos"),
                   dataTableOutput("articulos_DF")
            ),

            tabPanel("Prod Pais/año",
                   h3("Publicaciones de paises por año"),
                   plotOutput("paises_heatmap")
            ),
            tabPanel("Paises",
                   h3("Paises más productivos"),
                   plotOutput("paises_histograma")
            ),
            tabPanel("Colaboraciones",
                   h3("Colaboraciones entre paises"),
```

```
                plotOutput("colaboracion")
              )
            ),width=9
          )
),
tabPanel("Dosis utilizadas",
          sidebarPanel(
            selectInput(inputId= "selection_df1",
                        label="Elige localización:",
                        choices = localizacion_df),
            actionButton("update", "Cambiar"),
            hr(),

            sliderInput(inputId = "num2",
                        label = "Año",
                        value=c(2018), min=1980,max=2018)
          ),
          mainPanel(
            dataTableOutput("dosis")
          )
),
tabPanel("Wordcloud",
          sidebarPanel(
            selectInput(inputId= "selection",
                        label="Localización:",
                        choices = localizacion),
            actionButton("update", "Cambiar"),
            hr(),

            selectInput(inputId= "metodo_WC",
                        label="Método:",
                        choices = metodo),
            actionButton("update", "Cambiar"),
            hr(),

            sliderInput(inputId = "num",
                        label = "Elige un intervalo de tiempo",
                        value=c(2010,2014), min=1980,max=2018)
          ),
          mainPanel(
            plotOutput("wordcloud")
          )
),

tabPanel("Similitudes",
          sidebarPanel(
            selectInput(inputId= "selection1",
                        label="Elige localización:",
                        choices = localizacion),
            actionButton("update", "Cambiar"),
            hr(),

            sliderInput(inputId = "num1",
                        label = "Elige un año",
                        value=c(2018), min=1980,max=2018),

            sliderInput(inputId = "lim_weight",
                        label = "Nivel de correspondencia",
                        value=c(0.05), min=0,max=1)
          ),
          mainPanel(
            tabsetPanel(
              tabPanel("Similitud del coseno",
                       h3("Relaciones entre toxicidades y tratamientos por similitud del coseno"),
                       plotOutput("cluster")
              ),
              tabPanel("Similitud del coseno con walktraps",
```

```
                    h3("Relaciones entre toxicidades y tratamientos por
                      similitud del coseno agrupando en comunidades"),
                    plotOutput("cluster1")

                    )
          )
        )
        ),
  tabPanel("Clasificación en clusters",
        sidebarPanel(
          selectInput(inputId= "selection_df2",
                    label="Elige localización:",
                    choices = localizacion_df),
          actionButton("update", "Cambiar"),
          hr(),

          #selectInput(inputId= "metodo_cluster",
          #          label="Elige método:",
          #          choices = metodo_cl),
          #actionButton("update", "Cambiar"),
          #hr(),

          sliderInput(inputId = "numdf",
                    label = "Año",
                    value=c(2018), min=1980,max=2018),

          sliderInput(inputId = "k",
                    label = "Total de clusters",
                    value=c(6), min=1,max=10),

          sliderInput(inputId = "n_cluster",
                    label = "Selecciona cluster",
                    value=c(6), min=1,max=10)
        ),
        mainPanel(
          tabsetPanel(
            tabPanel("Dendograma",
                    h3("Dendograma"),
                    plotOutput("dendograma")
            ),
            tabPanel("Cluster plot",
                    h3("Abstracts en grupos"),
                    plotOutput("kmeans")
            ),
            tabPanel("Tabla Clusters",
                    h3("Términos que definen los Clusters"),
                    dataTableOutput("palabras_cl")
            ),
            tabPanel("Artículos clusters",
                    h3("Artículos de los distintos clusters"),
                    dataTableOutput("articulos_cl")
            )
          )
        )
  )
  ))

##########################################################################
#   SERVER
##########################################################################

server = function(input, output,session) {

  ################################################
  #  Función plot_Prod.bibliometrix
  ################################################
  plot_Prod.bibliometrix<-function(x, ...){
```

```
   if (class(x)!="bibliometrix"){cat('\n argument "x" have to be an object of class
"bibliometrix"\n');return(NA)}

   arguments <- list(...)
   if (sum(names(arguments)=="k")==0){k=10} else {k=arguments$k}
   if (sum(names(arguments)=="pause")==0){pause=FALSE} else {pause=arguments$pause}


   if (pause == TRUE){
     cat("Hit <Return> to see next plot: ")
     line <- readline()}


   Tab=table(x$Years)

   ## inserting missing years
   YY=setdiff(seq(min(x$Years),max(x$Years)),names(Tab))
   Y=data.frame(Year=as.numeric(c(names(Tab),YY)),Freq=c(as.numeric(Tab),rep(0,length(YY))))
   Y=Y[order(Y$Year),]

   names(Y)=c("Year","Freq")

   g=ggplot(Y, aes(x = Y$Year, y = Y$Freq)) +
     geom_line() +
     geom_area(fill = '#002F80', alpha = .5) +
     labs(x = 'Año'
          , y = 'Articulos'
          , title = "") +
     scale_x_continuous(breaks= (Y$Year[seq(1,length(Y$Year),by=5)])) +
     theme(text = element_text(color = "#444444")
           ,panel.background = element_rect(fill = '#EFEFEF')
           ,panel.grid.minor = element_line(color = '#FFFFFF')
           ,panel.grid.major = element_line(color = '#FFFFFF')
           ,plot.title = element_text(size = 24)
           ,axis.title = element_text(size = 18, color = '#555555')
           ,axis.title.y = element_text(vjust = 0.5, angle = 90)
           ,axis.title.x = element_text(hjust = 0.5)
     )
   plot(g)

 }

##############################################################################################
  ###################################
  #      Funcion plot_Autores
  ###################################
  plot_Autores<-function(x, ...){
    if (class(x)!="bibliometrix"){cat('\n argument "x" have to be an object of class
"bibliometrix"\n');return(NA)}

    arguments <- list(...)
    if (sum(names(arguments)=="k")==0){k=10} else {k=arguments$k}
    if (sum(names(arguments)=="pause")==0){pause=FALSE} else {pause=arguments$pause}

    if (pause == TRUE){
      cat("Hit <Return> to see next plot: ")
      line <- readline()}

    # Authors
    #barplot(x$Authors[1:k],horiz=TRUE,las=2,cex.names=0.5,main="Most Productive
Authors",xlab="Articles")
    xx=as.data.frame(x$Authors[1:k])
    g=ggplot(data=xx, aes(x=xx$AU, y=xx$Freq)) +
      geom_bar(stat="identity", fill="steelblue")+
      labs(title="", x = "Autores")+
      labs(y = "Nº de Documentos")+
```

```
    theme_minimal() +
    coord_flip()
  plot(g)
 }
 ##################################################################
 ########################################
 # FUNCION DOMINANCEL
 ########################################

 ## Identificar autores más frecuentes por tema.

 dominanceL<-function(results, k = 10){

  # Author Rank by Dominance Rank  (Kumar & Kumar, 2008)

  #options(warn=-1)

  if (class(results)!="bibliometrix"){cat('\n argument "results" have to be an object of class
"bibliometrix"\n');return(NA)}

  Nmf=table(results$FirstAuthors[results$nAUperPaper>1])
  FA=names(Nmf)
  #FA=gsub(" ", "", FA, fixed = TRUE)  # delete spaces

  AU=names(results$Authors)


  Mnt=rep(NA,k)
  for (i in 1:length(FA)){
   Mnt[i]=results$Authors[FA[i] == AU]

  }
  Dominance=round(Nmf/Mnt,2)

  t=0
  cont=0
  D=data.frame(matrix(NA,k,3))

  for (i in 1:length(FA)){
   if (sum(AU[i]==FA)>0){

    cont=cont+1
    D[cont,1]=Dominance[AU[i]==FA]
    D[cont,2]=results$Authors[i]
    D[cont,3]=Nmf[AU[i]==FA]

    row.names(D)[cont]=AU[i]
   }
   if (cont==k) break
  }


  D$RankbyArticles=1:dim(D)[1]
  D=D[order(-D[,2]),]
  D$RankDF=1:dim(D)[1]
  names(D)=c("Factor de Dominancia(DF)","Multi Autor","Autor Principal","Ranking  por
Articulos","Ranking por DF")
  return(D)
 }
 ##################################################
 # Funcion plot paises
 ##################################################
 plot_Paises.bibliometrix<-function(x, ...){

  if (class(x)!="bibliometrix"){cat('\n argument "x" have to be an object of class
"bibliometrix"\n');return(NA)}
```

```
  arguments <- list(...)
  if (sum(names(arguments)=="k")==0){k=10} else {k=arguments$k}
  if (sum(names(arguments)=="pause")==0){pause=FALSE} else {pause=arguments$pause}

  if (pause == TRUE){
    cat("Hit <Return> to see next plot: ")
    line <- readline()}
  if (!is.na(x$CountryCollaboration[1,1])){
    # Countries
    xx=x$CountryCollaboration[1:k,]
    xx=xx[order(-(xx$SCP+xx$MCP)),]
    xx1=cbind(xx[,1:2],rep("SCP",k))
    names(xx1)=c("Country","Freq","Collaboration")
    xx2=cbind(xx[,c(1,3)],rep("MCP",k))
    names(xx2)=c("Country","Freq","Collaboration")
    xx=rbind(xx2,xx1)
    xx$Country=factor(xx$Country,levels=xx$Country[1:dim(xx2)[1]])
    g=suppressWarnings(ggplot(data=xx, aes(x=xx$Country, y=xx$Freq,fill=xx$Collaboration)) +
                        geom_bar(stat="identity")+
                        scale_x_discrete(limits = rev(levels(xx$Country)))+
                        scale_fill_discrete(name="Colaboración",
                                            breaks=c("SCP","MCP"))+
                        labs(title = "Paises más Productivos", x = "Paises", y = "Nº de Documentos",
                             caption = "SCP: Single Country Publications, MCP: Multiple Country Publications")+
                        theme_minimal() +
                        theme(plot.caption = element_text(size = 9, hjust = 0.5,
                                                          color = "blue", face = "italic"))+
                        coord_flip())
    plot(g)
  }
}
##################################
# Funcion Summary
##################################

summary.bibliometrix<-function(object,k){
  if (!is.null(object$ID) & !is.null(object$DE)){

    AAA=data.frame(cbind(object$DE[1:k]))
    AAA$MPA=row.names(AAA);AAA=AAA[,c(2,1)]
    names(AAA)=c("word", "freq")

    return(AAA)
  }
}

#####################################################################


df_tox_rad_G <- reactive({read.table(sprintf("%s", input$selection_df), header = T, sep = ";",
                                     na.strings = "NA",colClasses = "character")
})


results <- reactive({biblioAnalysis(df_tox_rad_G(), sep = ";")})

DF <- reactive({dominanceL(results(), k = 10)})

ranking_AU <<- reactive({rownames(DF())})

observeEvent(ranking_AU(), {
  updateSelectInput(session,"selection_AU", choices = ranking_AU())
},once=F)

observeEvent(input$k, {
  updateSliderInput(session,"n_cluster", max = input$k)
},once=F)
```

```
Meta <- reactive({metaTagExtraction(df_tox_rad_G(), Field = "AU_CO", sep = ";")})

##################################################################
# Para la Wordcloud

abstract <- reactive({readabs(sprintf("%s.txt",input$selection)) })
abstracts <- reactive({searchabsL(abstract(),yr=input$num) })

#Elaboramos la taxonomia de las toxicidades con ayuda del artículo de Emami
# "Tolerance of Normal Tissue to Therapeutic Radiation"

tdm_tox <- c("neuritis","hearing loss", "cranial neuropathy","myelitis","neuropathy","stenosis",
          "fistula","pericarditis","long-term cardiac mortality", "aneurysim","ulceration",
          "enteritis","obstruction","colitis","proctitis","impotence","necrosis",
          "malignant hypertension","pneumonitis","neurologic toxicity","paralysis",
          "sensory deficits","pain","bowel incontinence","bladder incontinence", "visual loss",
          "RION", "Radiation-induced optic neuropathy", "blindness","radiation retinopathy",
          "loss of visual acuity","sensorineural hearing loss", "SNHL", "salivary dydfunction",
          "Xerostomia", "osteoradionecrosis", "ORN","dysphagia", "aspiration","vocal dysfunction",
          "laryngeal edema","paresthesia","radiation pneumonitis","RP","esophagitis",
          "Radiation-induced liver disease","RIDL", "Radiation-induce renal dysfunction",
          "renal failure","dyspepsia","ulceracion", "GU","GI")


#Creamos la lista de tipos de tratamiento

tdm_tra <-
c("3D","IMRT","SBRT","VMAT","Tomotherapy","Brachytherapy","Immunotherapy","Conventional",
"Conformal",
          "Hypofractionated","Protontherapy", "CRT", "Conformal radiotherapy","SABR","hydrogel",
          "rectal balloon","metastasis", "oligometastasis","oligometastatic","seed","seeds", "IGRT")


tdm_tox_tra <- reactive({c(tdm_tox,tdm_tra)})

tdmA <- reactive({tdm_for_lsa(abstracts(),tdm_tox_tra())})

####

df_tox_rad <- reactive({read.table(sprintf("df_%s", input$selection), header = T, sep = ";",
                           na.strings = "NA",colClasses = "character")
})

PTR_WC1 <- reactive({df_tox_rad()[,c(2,3,4)]})

PTR_WC <- reactive({PTR_WC1()[PTR_WC1()$PY==input$num,]})

corpusWC1 <- reactive({Corpus(VectorSource(PTR_WC()[,2]),
                        readerControl = list(reader=readPlain,language= "en", load = T))})

corpusWC2 <- reactive({tm_map(corpusWC1(), removeNumbers)})
corpusWC3 <- reactive({tm_map(corpusWC2(), removePunctuation)})
corpusWC4 <- reactive({tm_map(corpusWC3(), tolower)})
corpusWC5 <- reactive({tm_map(corpusWC4(), removeWords, stopwords("english"))})
corpusWC6 <- reactive({tm_map(corpusWC5(), stemDocument, language="english")})
corpusWC <- reactive({tm_map(corpusWC6(), stripWhitespace)})

# Utilizamos el paquete tm para calcular el dtm
dtm_WC <- reactive({DocumentTermMatrix(corpusWC()) })

#Lo mismo pero en formato de datos
dtm.data_WC <- reactive({as.matrix(dtm_WC())})
####

df_tox_rad_yr <- reactive({df_tox_rad()[df_tox_rad()$PY==input$num,]})
```

```
results1 <- reactive({biblioAnalysis(df_tox_rad_yr(), sep = ";")})

sumario <- reactive({summary.bibliometrix(results1(), k = 150)})


#########################################################################
# Para el cálculo de similitudes

abstract1 <- reactive({readabs(sprintf("%s.txt", input$selection1)) })
tdmA1 <- reactive({tdm_for_lsa(abstracts1(),tdm_tox_tra())})
abstracts1 <- reactive({searchabsL(abstract1(),yr=input$num1) })

#########################################################################
# Para separar los abstrtacts en clusters

#df_tox_rad <- df_tox_rad1()[df_tox_rad1()$AB!="",]

df_prost_tox_rad <- reactive({read.table(sprintf("%s", input$selection_df2), header = T, sep = ";",
                                na.strings = "NA",colClasses = "character")
})

PTR_class1 <- reactive({df_prost_tox_rad()[df_prost_tox_rad()$AB!="",c(2,3,4,20,7)]})

#Utilizo solo los de un año para no tener tantos datos

PTR_class <- reactive({PTR_class1()[PTR_class1()$PY==input$numdf,]})

# se crea el documento matrix-término

dtm <- reactive({CreateDtm(doc_vec = PTR_class()$AB,
                        doc_names = PTR_class()$UT,
                        ngram_window = c(1, 2),
                        stopword_vec = stopwords::stopwords("en"),
                        lower = TRUE,
                        remove_punctuation = TRUE,
                        remove_numbers = TRUE,
                        verbose = FALSE,
                        cpus = 2)
})

# se construye la matriz témino-documento

tf_mat <- reactive({TermDocFreq(dtm())})

# TF-IDF similaridad del coseno
tfidf1 <- reactive({t(dtm()[ , tf_mat()$term ]) * tf_mat()$idf})

tfidf <- reactive({t(tfidf1())})
csim1 <- reactive({tfidf() / sqrt(rowSums(tfidf() * tfidf()))})

csim <- reactive({csim1() %*% t(csim1())})

cdist <- reactive({as.dist(1 - csim())})

hc <- reactive({ hclust(cdist(), "ward.D")})

clustering <- reactive({cutree(hc(), input$k)})



#########################################################################################
#########
 df_tox_rad1 <- reactive({read.table(sprintf("%s", input$selection_df1), header = T, sep = ";",
                               na.strings = "NA",colClasses = "character")
 })
```

```
##########################################
# OUTPUTS
##########################################

output$produccion <- renderPlot({


  plot_Prod.bibliometrix(x = results(), k = 10, pause = FALSE)

},height = 600)
#####################################################################

output$autores <- renderPlot({

  plot_Autores(x = results(), k = 10, pause = FALSE)

},height = 600)

#####################################################################

output$dominancia <- DT::renderDataTable({

  #DF <- dominanceL(results(), k = 10)

  DT::datatable(DF())

})
#####################################################################

output$articulos_DF <- DT::renderDataTable({


  df_AU_TC <-df_tox_rad_G()[regexpr(input$selection_AU,df_tox_rad_G()$AU)>0,]


  df_AU_TC <- df_AU_TC[,c(20,2,4,7)]
  df_AU_TC$TC <- as.integer(df_AU_TC$TC)

  df_AU_TC$UT <- paste0("<a href=https://www.ncbi.nlm.nih.gov/pubmed/?term=",
                        df_AU_TC$UT," target='_blank'>",df_AU_TC$UT,"</a>")

  names(df_AU_TC)<-c("PMID","Título","Año","Citas")

  DT::datatable(df_AU_TC,rownames = F,escape = F,
                caption = sprintf("Articulos de %s",input$selection_AU))


})
#####################################################################

output$articulos <- DT::renderDataTable({

  df_TC <-df_tox_rad_G()[,]
  df_TC_2 <- df_TC[,c(20,2,4,7)]
  df_TC_2$TC <- as.integer(df_TC_2$TC)
  df_TC_2$UT <- paste0("<a href=https://www.ncbi.nlm.nih.gov/pubmed/?term=",
                       df_TC_2$UT," target='_blank'>",df_TC_2$UT,"</a>")

  names(df_TC_2)<-c("PMID","Título","Año","Citas")

  DT::datatable(df_TC_2,rownames = F,escape = F)

})

#####################################################################
```

```
output$paises_heatmap <- renderPlot({

  art_co_yr <- df_tox_rad_G()[,c(21,4)]

  art_co_yr=art_co_yr[order(art_co_yr$AU_CO,decreasing = T),]


  tabla <- table(art_co_yr$PY,art_co_yr$AU_CO)

  heatmap.2(tabla,
            dendrogram='none',
            Colv = F, Rowv = F,
            density.info='none',
            trace='none',
            col = colorRampPalette(c('black','green','yellow','red'))(50))


},height = 600)

#####################################################################

output$paises_histograma <- renderPlot({

  plot_Paises.bibliometrix(x = results(), k = 10, pause = FALSE)

},height = 600)
#####################################################################

output$colaboracion <- renderPlot({

  NetMatrix <- biblioNetwork(Meta(), analysis = "collaboration", network = "countries",
                             sep = ";")


  net=networkPlot(NetMatrix, n = dim(NetMatrix)[1], Title = "",
                  type = "circle", size=TRUE, remove.multiple=FALSE,labelsize=1.2,
                  cluster="none")
},height = 650)

#####################################################################

output$dosis <- DT::renderDataTable({

  #Quito las filas donde la celda del abstracts está vacia

  df_tox_rad <- df_tox_rad1()[df_tox_rad1()$AB!="",]

  # Me voy a quedar con los artículos que en el abstract tengan"Gy"
  anyo <- input$num2
  df_AB_Gy <-df_tox_rad[regexpr("GY",df_tox_rad$AB)>0,]
  df_AB_Gy <-df_AB_Gy[df_AB_Gy$PY==anyo,]
  df_AB_Gy <-df_AB_Gy[regexpr("DOSE",df_AB_Gy$AB)>0,]
  df_AB_Gy <- df_AB_Gy[,c(2,3,20,4,7)]

  #Separo las frases de los abstracts:

  y<-""
  nrows <- nrow(df_AB_Gy)
  df_AB_Gy_Token <-matrix(y,nrow=nrow(df_AB_Gy),ncol=nrow(df_AB_Gy))

  for (i in 1:nrows)
  {
    token <- SentenceToken(df_AB_Gy[i,2])
    trows <- length(token)
    for (j in 1:trows)
    {
      df_AB_Gy_Token[i,j] <- t(token[j])
```

```
    }
   }
   rownames(df_AB_Gy_Token) <- df_AB_Gy[,3]

   # Me quedo con las frases que contengan "GY":


   for (i in 1:nrows)
   {
    for (j in 1:trows)
    {
     if (regexpr("GY",df_AB_Gy_Token[i,j])>0)
     {df_AB_Gy_Token[i,j] <- df_AB_Gy_Token[i,j]}
     else {df_AB_Gy_Token[i,j]=""}
    }
   }

   # Ahora vuelvo a juntar las frases:

   Resumen <- paste(df_AB_Gy_Token[,1],df_AB_Gy_Token[,2],df_AB_Gy_Token[,3],
                    df_AB_Gy_Token[,4],df_AB_Gy_Token[,5],df_AB_Gy_Token[,6],
                    df_AB_Gy_Token[,7],df_AB_Gy_Token[,8],df_AB_Gy_Token[,9],
                    df_AB_Gy_Token[,10],df_AB_Gy_Token[,11],df_AB_Gy_Token[,12],
                    df_AB_Gy_Token[,13],df_AB_Gy_Token[,14],df_AB_Gy_Token[,15],sep="")

   PMID <- df_AB_Gy[,3]
   Año <- df_AB_Gy[,4]
   Citas <- df_AB_Gy[,5]
   resu <- cbind(PMID,Resumen,Año,Citas)
   resu <- as.data.frame(resu)
   resu$Citas <- as.integer(resu$Citas)
   resu[,1] <- paste0("<a href=https://www.ncbi.nlm.nih.gov/pubmed/?term=",resu[,1],"
target='_blank'>",resu[,1],"</a>")

   names(resu)<-c("PMID","Resumen","Año","Citas")

   DT::datatable(resu,rownames = F,escape = F,
                 options = list(searchHighlight = TRUE, search = list(search = 'GY')))

 })

 ####################################################
 output$wordcloud <- renderPlot({

   #Tenemos tres opciones para elaborar la wordcloud

   if (input$metodo_WC == "1") {
    m <- as.matrix(tdmA())
    v <- sort(rowSums(m),decreasing=TRUE)
    d <- data.frame(word = names(v),freq=v)

    set.seed(131)
    wordcloud(words = d$word, freq = d$freq, min.freq = 1,scale=c(8,0.8),
              max.words=150, random.order=FALSE, rot.per=0.35,
              colors=brewer.pal(8, "Dark2"))
   }
   if (input$metodo_WC == "2")  {
    m <- as.matrix(t(dtm.data_WC()))
    v <- sort(rowSums(m),decreasing=TRUE)
    d <- data.frame(word = names(v),freq=v)

    set.seed(131)
    wordcloud(words = d$word, freq = d$freq, min.freq = 1,scale=c(8,0.8),
              max.words=150, random.order=FALSE, rot.per=0.35,
              colors=brewer.pal(8, "Dark2"))
   }
```

```
  if (input$metodo_WC == "3")  {
    d <- sumario()

    set.seed(131)
    wordcloud(words = d$word, freq = d$freq, min.freq = 1,scale=c(4,0.4),
              max.words=150, random.order=FALSE, rot.per=0.35,
              colors=brewer.pal(8, "Dark2"))}

},height = 650)

#######################################################
output$cluster <- renderPlot({

  cos_sim_calc(tdmA1())
  cos=read.table("cossimdata.txt",header=FALSE,sep="\t")

  cos=na.omit(cos)
  cos1=cos[cos$V3>=0.01,]

  set.seed(131)
  relations <- data.frame(from=cos[,1], to=cos[,2],   weight=abs(cos[,3]))
  #quitamos los ceros
  relations2=relations[-row(relations)[relations == 0],]

  relations2=relations2[relations2$weight>input$lim_weight,]
  g.1a <- graph.data.frame(relations2, directed=FALSE)
  V(g.1a)$size<-6
  min<-.95
  layout1 <- layout.auto(g.1a)

  opar <- par()$mar; par(mar=rep(3, 4))
  plot(g.1a, layout=layout1)

},height = 650)
#######################################################
output$cluster1 <- renderPlot({

  cos_sim_calc(tdmA1())
  cos=read.table("cossimdata.txt",header=FALSE,sep="\t")

  cos=na.omit(cos)
  cos1=cos[cos$V3>=0.01,]

  set.seed(131)
  relations <- data.frame(from=cos[,1], to=cos[,2],   weight=abs(cos[,3]))
  # quitamos los ceros
  relations2=relations[-row(relations)[relations == 0],]

  relations2=relations2[relations2$weight>input$lim_weight,]
  g.1a <- graph.data.frame(relations2, directed=FALSE)
  V(g.1a)$size<-6
  min<-.95
  layout1 <- layout.auto(g.1a)

  opar <- par()$mar; par(mar=rep(3, 4))

  wc=cluster_walktrap(g.1a)
  modularity(wc)

  plot(wc,g.1a, layout=layout1)
},height = 650)
#######################################################
output$dendograma <- renderPlot({

  plot(hc(),main = sprintf ("Dendograma año %s",input$numdf))
  rect.hclust(hc(), k = input$k)
```

```
  },height = 650)
  ########################################################
  output$kmeans <- renderPlot({

    clustering <- reactive({cutree(hc(), input$k)})

    fviz_cluster(list(data = cdist(), cluster = clustering()))
  },height = 650)

  ########################################################
  output$palabras_cl <- DT::renderDataTable({



    p_words <- colSums(dtm()) / sum(dtm())

    cluster_words <- lapply(unique(clustering()), function(x){
      rows <- dtm()[ clustering() == x , ]

      rows <- rows[ , colSums(rows) > 0 ]

      colSums(rows) / sum(rows) - p_words[ colnames(rows) ]
    })

    # creamos una tabla con los 5 términos que definen cada cluster
    cluster_summary <- data.frame(cluster = unique(clustering()),
                                  tamaño = as.numeric(table(clustering())),
                                  términos = sapply(cluster_words, function(d){
                                    paste(
                                      names(d)[ order(d, decreasing = TRUE) ][ 1:5 ],
                                      collapse = ", ")
                                  }),
                                  stringsAsFactors = FALSE)

    DT::datatable(cluster_summary,rownames = F,escape = F)

  })
  ########################################################
  output$articulos_cl <- DT::renderDataTable({

    PTR_class_cluster2 <- cbind(PTR_class()[,c(4,1)],as.numeric(clustering()))

    colnames(PTR_class_cluster2) <- c("PMID","Titulo","Clustering")
    as.data.frame(PTR_class_cluster2)

    cluster_n <- PTR_class_cluster2[PTR_class_cluster2[,3] == input$n_cluster,]

    cluster_n[,1] <- paste0("<a href=https://www.ncbi.nlm.nih.gov/pubmed/?term=",cluster_n[,1],"
target='_blank'>",cluster_n[,1],"</a>")

    elementos <- nrow(cluster_n)

    DT::datatable(cluster_n[,c(1,2)],rownames = F,escape = F,
                  caption = sprintf("Articulos del cluster %s con %s elementos",
                                    input$n_cluster,elementos))

  })
  ########################################################
}
shinyApp(ui=ui,server=server)
```