

Package: diariosuy (via r-universe)

October 6, 2026

Title Buscador y Analizador de Diarios de Sesión del Parlamento Uruguayo

Version 0.1.0

Description Busca sesiones parlamentarias por palabra clave en el sitio del Parlamento del Uruguay, descarga los diarios de sesión en PDF y analiza sus textos: frecuencia de términos, evolución en el tiempo y co-ocurrencias. La app se ejecuta en la máquina del usuario, condición necesaria para consultar el sitio oficial, que rechaza las conexiones desde fuera de Uruguay.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Depends R (>= 4.1)

Imports bslib, dplyr, DT, fs, ggplot2, httr, pdftools, plotly, purrr, rlang, rvest, shiny, shinyFiles, stringr, tibble, tidyr, xml2, zip

URL <https://github.com/EmilianoGonLas/diariosuy>

BugReports <https://github.com/EmilianoGonLas/diariosuy/issues>

Config/pak/sysreqs cmake make libicu-dev libjpeg-dev libuv1-dev libxml2-dev libssl-dev libpoppler-cpp-dev poppler-data zlib1g-dev

Repository <https://emilianogonlas.r-universe.dev>

Date/Publication 2026-09-06 16:41:26 UTC

RemoteUrl <https://github.com/EmilianoGonLas/diariosuy>

RemoteRef master

RemoteSha 7c18f80cf1fb423de37baa6f3bf8061a8e3bd834

Contents

.contar_proximos	2
.extraer_fragmento	3
.extraer_texto_pdf	3
.reconstruir_columna	4
abrir_app	4
analizar_pdfs	5
buscar_sesiones	6
crear_zip	6
descargar_sesiones	7
estrellas_html	7
grafico_coocurrencias	8
grafico_timeline	8
LEGISLATURAS	9
obtener_url_pdf	9
Index	10

<code>.contar_proximos</code>	<i>Cuenta cuántas menciones de patron1 tienen a patron2 dentro de ventana caracteres en el mismo texto (co-ocurrencia de proximidad).</i>
-------------------------------	--

Description

Cuenta cuántas menciones de **patron1** tienen a **patron2** dentro de **ventana** caracteres en el mismo texto (co-ocurrencia de proximidad).

Usage

```
.contar_proximos(texto, patron1, patron2, ventana = 500L)
```

Arguments

<code>texto</code>	Texto normalizado (minúsculas, espacios simples)
<code>patron1</code>	Regex del término principal
<code>patron2</code>	Regex del término de cruce
<code>ventana</code>	Máx. distancia en caracteres para considerar "cerca" (default 500)

Value

Entero: cuántas menciones de **patron1** tienen a **patron2** "cerca"

<code>.extraer_fragmento</code>	<i>Extrae un fragmento de texto que empiece al inicio de una oración.</i>
---------------------------------	---

Description

Busca el último signo de puntuación en los `max_pre` caracteres anteriores al match y empieza allí. Si no hay ninguno, toma desde el límite. Cierra en el primer punto dentro de los `max_total` caracteres siguientes.

Usage

```
.extraer_fragmento(texto, s, e, max_pre = 90L, max_total = 230L)
```

Arguments

<code>texto</code>	Texto normalizado
<code>s, e</code>	Inicio y fin del match a destacar
<code>max_pre</code>	Máx. chars a buscar hacia atrás para inicio de oración (default 90)
<code>max_total</code>	Longitud máxima del fragmento (default 230)

Value

Lista con: ws (índice inicio), we (índice fin), text (el fragmento)

<code>.extraer_texto_pdf</code>	<i>Extrae texto de un PDF respetando el orden de lectura en documentos a dos columnas.</i>
---------------------------------	--

Description

Usa `pdftools::pdf_data()` para obtener las coordenadas de cada palabra y reconstruir el texto leyendo primero la columna izquierda completa y luego la derecha, en lugar de ir línea a línea de izquierda a derecha. Si `pdf_data()` falla, cae en `pdf_text()` como fallback.

Usage

```
.extraer_texto_pdf(ruta)
```

Arguments

<code>ruta</code>	Ruta al archivo PDF
-------------------	---------------------

Value

Character vector con el texto de cada página (largo = n° de páginas)

<code>.reconstruir_columna</code>	<i>Reconstruye el texto de una columna agrupando palabras en líneas.</i>
-----------------------------------	--

Description

Ordena por (y, x), detecta saltos de línea por el Δy entre palabras consecutivas y une las palabras de cada línea con espacios.

Usage

```
.reconstruir_columna(datos)
```

Arguments

<code>datos</code>	data.frame con columnas x, y, width, height, text (de pdf_data())
--------------------	---

Value

String con el texto de la columna en orden de lectura correcto

<code>abrir_app</code>	<i>Abrir la app de diarios del Parlamento</i>
------------------------	---

Description

Levanta la aplicación Shiny y la abre en el navegador. Todas las consultas salen desde la máquina donde se ejecuta, que es lo que permite llegar al sitio del Parlamento: `parlamento.gub.uy` responde 403 a las IP de fuera de Uruguay, así que la app necesita correr desde una conexión uruguaya.

Usage

```
abrir_app(puerto = NULL, navegador = TRUE)
```

Arguments

<code>puerto</code>	Puerto donde levantar la app. Por defecto Shiny elige uno libre.
<code>navegador</code>	Si es TRUE (por defecto) abre el navegador automáticamente.

Value

Se invoca por su efecto: corre la app hasta que se la interrumpe.

Examples

```
if (interactive()) {  
  abrir_app()  
}
```

analizar_pdfs	<i>Analiza PDFs buscando un término principal y co-ocurrencias opcionales</i>
----------------------	---

Description

Analiza PDFs buscando un término principal y co-ocurrencias opcionales

Usage

```
analizar_pdfs(  
  directorio,  
  termino_principal,  
  terminos_cruce = list(),  
  progreso_fn = NULL  
)
```

Arguments

directorio	Carpeta con los PDFs a analizar. También acepta la ruta de un archivo .zip: en ese caso se extrae a una carpeta temporal y se analiza su contenido, así el ZIP que arma el Buscador se puede usar tal cual. La búsqueda de PDFs es recursiva, de modo que no importa si quedaron dentro de subcarpetas.
termino_principal	Expresión regular del término principal
terminos_cruce	Lista nombrada: <code>list(turismo = "turist extranjer")</code>
progreso_fn	<code>function(valor, mensaje)</code> para reportar progreso

Value

tibble con columnas: `archivo`, `ruta`, `legislatura`, `fecha`, `id_doc`, `paginas_con_termino`, `cruce_[nombre]`, `total_cruces`, `puntuacion`, `fragmentos_texto`, `estado`

buscar_sesiones	<i>Busca sesiones parlamentarias que mencionan un término dado</i>
-----------------	--

Description

Busca sesiones parlamentarias que mencionan un término dado

Usage

```
buscar_sesiones(texto, camara = "All", lgl_ids = c(49, 50), progreso_fn = NULL)
```

Arguments

texto	Término a buscar (ej: "CANNABIS", "VIVIENDA SOCIAL")
camara	Cámara: "All" (ambas), "S" (Senado), "R" (Representantes)
lgl_ids	Vector de IDs de legislatura a buscar (ver LEGISLATURAS en global.R)
progreso_fn	Función opcional para reportar progreso: function(valor, mensaje)

Value

tibble con columnas: legislatura, lgl_id, cuerpo, fecha, id_doc, url_intermedia. Lleva siempre el atributo "errores": vector de mensajes con los fallos de red/HTTP que hubo. Un tibble vacío CON errores significa "no se pudo consultar el sitio"; vacío SIN errores, "el sitio no devolvió nada".

crear_zip	<i>Empaqueta todos los PDFs de una carpeta en un archivo ZIP</i>
-----------	--

Description

Usa nombres de archivo relativos para garantizar que el ZIP se pueda abrir correctamente en cualquier sistema operativo.

Usage

```
crear_zip(directorio, destino)
```

Arguments

directorio	Carpeta con los PDFs a empaquetar
destino	Ruta del archivo ZIP de salida

Value

TRUE si el ZIP se creó correctamente, FALSE si no había PDFs

<code>descargar_sesiones</code>	<i>Descarga PDFs de sesiones seleccionadas a un directorio local</i>
---------------------------------	--

Description

Cada PDF se nombra con el patrón: `Lgl_[LEGISLATURA]_[FECHA]_ID_[ID].pdf` Si el archivo ya existe, lo omite sin re-descargarlo.

Usage

```
descargar_sesiones(sesiones, directorio, progreso_fn = NULL)
```

Arguments

<code>sesiones</code>	tibble con columnas: legislatura, fecha, id_doc, url_intermedia
<code>directorio</code>	Carpeta destino (se crea si no existe)
<code>progreso_fn</code>	Función opcional: <code>function(valor, mensaje)</code> para reportar progreso

Value

tibble con columnas: archivo, estado (resultado por cada PDF)

<code>estrellas_html</code>	<i>Convierte score_raw a estrellas visuales ()</i>
-----------------------------	---

Description

Convierte `score_raw` a estrellas visuales ()

Usage

```
estrellas_html(score_raw, max_stars = 5L)
```

Arguments

<code>score_raw</code>	Vector numérico de scores crudos
<code>max_stars</code>	Número máximo de estrellas (default 5)

Value

Vector de strings HTML con estrellas

grafico_coocurrencias

Gráfico de co-ocurrencias entre el término principal y términos secundarios

Description

Gráfico de co-ocurrencias entre el término principal y términos secundarios

Usage

```
grafico_coocurrencias(datos, terminos_cruce)
```

Arguments

datos tibble resultado de analizar__pdfs()
terminos_cruce lista nombrada usada en el análisis

Value

objeto plotly o NULL si no hay términos de cruce

grafico_timeline

Gráfico interactivo de menciones a lo largo del tiempo

Description

Gráfico interactivo de menciones a lo largo del tiempo

Usage

```
grafico_timeline(datos, titulo = "Menciones del término en el tiempo")
```

Arguments

datos tibble resultado de analizar__pdfs()
titulo Título del gráfico

Value

objeto plotly

LEGISLATURAS*Legislaturas disponibles en el buscador del Parlamento*

Description

Tabla con las legislaturas que se pueden consultar, con el rango de fechas que el buscador oficial espera para cada una.

Usage

LEGISLATURAS

Format

Un tibble con una fila por legislatura y las columnas `etiqueta`, `lgl_id`, `fecha_desde` y `fecha_hasta`.

Details

Los diarios de sesión están disponibles en línea a partir del 15/02/1985 (inicio de la XLII, restauración democrática). Las legislaturas XL y XLI son previas al golpe de Estado de 1973: pueden tener documentos limitados o ninguno según el cuerpo que se busque.

obtener_url_pdf*Obtiene la URL directa del PDF desde la página intermedia del parlamento*

Description

Usa `httr` con User-Agent de navegador real. Prueba tres selectores CSS en cascada para cubrir tanto la estructura actual del sitio como páginas antiguas de legislaturas anteriores (que usan HTML diferente).

Usage

```
obtener_url_pdf(url_intermedia)
```

Arguments

`url_intermedia`

URL del visor de documentos del parlamento

Value

URL del PDF como string, o NA si no se encuentra

Index

- * datasets
 - LEGISLATURAS, [9](#)
 - .contar_proximos, [2](#)
 - .extraer_fragmento, [3](#)
 - .extraer_texto_pdf, [3](#)
 - .reconstruir_columna, [4](#)
- abrir_app, [4](#)
- analizar_pdfs, [5](#)
- buscar_sesiones, [6](#)
- crear_zip, [6](#)
- descargar_sesiones, [7](#)
- estrellas_html, [7](#)
- grafico_coocurrencias, [8](#)
- grafico_timeline, [8](#)
- LEGISLATURAS, [9](#)
- obtener_url_pdf, [9](#)