

# Package ‘ACEP’

October 19, 2023

**Type** Package

**Title** Analisis Computacional de Eventos de Protesta

**Version** 0.0.22

**Description** La libreria 'ACEP' contiene funciones especificas para desarrollar analisis computacional de eventos de protesta. Asimismo, contiene base de datos con colecciones de notas sobre protestas y diccionarios de palabras conflictivas. Coleccion de diccionarios que reúne diccionarios de diferentes origenes. The 'ACEP' library contains specific functions to perform computational analysis of protest events. It also contains a database with collections of notes on protests and dictionaries of conflicting words. Collection of dictionaries that brings together dictionaries from different sources.

**License** MIT + file LICENSE

**URL** <https://github.com/agusnieto77/ACEP>,  
<https://agusnieto77.github.io/ACEP/>

**BugReports** <https://github.com/agusnieto77/ACEP/issues>

**Depends** R (>= 3.5.0)

**Imports** graphics, stats, httr

**Suggests** covr, testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Encoding** UTF-8

**Language** es

**LazyData** true

**RoxygenNote** 7.2.1

**NeedsCompilation** no

**Author** Agustin Nieto [aut, cre] (<<https://orcid.org/0000-0003-4467-873X>>)

**Maintainer** Agustin Nieto <agustin.nieto77@gmail.com>

**Repository** CRAN

**Date/Publication** 2023-10-19 07:00:23 UTC

## R topics documented:

acep_bases	2
acep_clean	3
acep_db	4
acep_detect	5
acep_diccionarios	6
acep_frec	7
acep_int	7
acep_load_base	8
acep_men	8
acep_plot_rst	9
acep_plot_st	10
acep_rs	11
acep_rst	11
acep_token	12
acep_token_plot	13
acep_token_table	14
<b>Index</b>	<b>15</b>

---

acep\_bases *Coleccion de notas.*

---

### Description

Contiene colecciones de notas de distintos portales noticiosos (una muestra corta). Una segunda coleccion es de notas del periodico bahiense La Nueva. Tambien tiene resúmenes estadísticos de las bases completas para el desarrollo de los ejemplos de las funciones.

### Usage

```
data(acep_bases)
```

### Format

Es una lista con 8 objetos.

**la\_nueva** es un data frame con notas de La Nueva

**rev\_puerto** es un data frame con notas de la Revista Puerto

**rp\_procesada** es un data frame con indicadores de conflictividad basados en los datos de la Revista Puerto

**lc\_mdp** es una url para la descarga del corpus de notas de La Capital

**rp\_mdp** es una url para la descarga del corpus de notas de la Revista Puerto

**ed\_neco** es una url para la descarga del corpus de notas de Ecos Diarios

**ln\_bb** es una url para la descarga del corpus de notas de La Nueva

**ln\_arg** es una url para la descarga del corpus de notas de La Nacion

**Source**

Revista Puerto

La Nueva

**References**

Nieto, Agustin 2020 «Intersecciones entre historia digital e historia social: un ejercicio de lectura distante sobre la conflictividad marítima en la historia argentina reciente». Drassana: revista del Museu Maritim (28):122-42. (Revista Drassana)

**Examples**

```
acep_bases$rp_procesada[1:6, ]
```

---

acep\_clean

*Limpieza de texto.*

---

**Description**

Funcion que limpia y normaliza las notas/textos.

**Usage**

```
acep_clean(  
  x,  
  tolower = TRUE,  
  rm_esp = TRUE,  
  rm_emoji = TRUE,  
  rm_hashtag = TRUE,  
  rm_users = TRUE,  
  rm_punt = TRUE,  
  rm_num = TRUE,  
  rm_url = TRUE,  
  rm_meses = TRUE,  
  rm_dias = TRUE,  
  rm_stopwords = TRUE,  
  rm_shortwords = TRUE,  
  rm_newline = TRUE,  
  rm_whitespace = TRUE,  
  other_sw = NULL,  
  u = 1  
)
```

**Arguments**

x	vector de textos al que se le aplica la funcion de limpieza de texto.
tolower	convierte los textos a minusculas.
rm_cesp	remueve caracteres especiales.
rm_emoji	remueve los emojis.
rm_hashtag	remueve los hashtags.
rm_users	remueve las menciones de usuarios de redes sociales.
rm_punt	remueve la puntuacion.
rm_num	remueve numeros.
rm_url	remueve las url.
rm_meses	remueve los meses del anio.
rm_dias	remueve los dias de la semana.
rm_stopwords	remueve palabras vacias.
rm_shortwords	remueve las palabras cortas.
rm_newline	remueve los saltos de linea.
rm_whitespace	remueve los espacios en blanco.
other_sw	su valor por defecto es NULL, sirve para ampliar el listado de stopwords con un nuevo vector de palabras.
u	umbral de caracteres para la funcion rm_shortwords.

**Examples**

```
acep_clean("El SUTEBA fue al paro. Reclaman mejoras salariales.", rm_cesp = FALSE)
```

---

acep\_db

*Frecuencia, menciones e intensidad.*

---

**Description**

Funcion que usa las funciones `acep_freq`, `acep_men` y `acep_int` y devuelve una tabla con tres columnas nuevas: numero de palabras, numero de menciones del diccionario, indice de intensidad.

**Usage**

```
acep_db(db, t, d, n)
```

**Arguments**

db	data frame con los textos a procesar.
t	columna de data frame que contiene el vector de textos a procesar.
d	diccionario en formato vector.
n	cantidad de decimales del indice de intensidad.

**Value**

Si todas las entradas son correctas, la salida sera una base de datos en formato tabular con tres nuevas variables.

**Examples**

```
df <- data.frame(texto = c("El SUTEBA fue al paro. Reclaman mejoras salariales.",
"El SOIP lleva adelante un plan de lucha con paros y piquetes."))
diccionario <- c("paro", "lucha", "piquetes")
acep_db(df, df$texto, diccionario, 4)
```

---

acep\_detect

*Deteccion de menciones de palabras.*

---

**Description**

Funcion que detecta de menciones de palabras que refieren a conflictos en cada una de las notas/textos.

**Usage**

```
acep_detect(x, y, u = 1, tolower = TRUE)
```

**Arguments**

x	vector de textos al que se le aplica la funcion de deteccion de menciones de palabras del diccionario.
y	vector de palabras del diccionario utilizado.
u	umbral para atribuir valor positivo a la deteccion de las menciones.
tolower	convierte los textos a minusculas.

**Examples**

```
df <- data.frame(texto = c("El SUTEBA fue al paro. Reclaman mejoras salariales.",
"El SOIP lleva adelante un plan de lucha con paros y piquetes."))
diccionario <- c("paro", "lucha", "piquetes")
df$detect <- acep_detect(df$texto, diccionario)
df
```

---

acep\_diccionarios      *Coleccion de diccionarios.*

---

### Description

Coleccion de diccionarios que reúne diccionarios de diferentes orígenes. El diccionario `dicc_confl_acep` fueron construidos en el marco del Observatorio de Conflictividad de la UNMDP. Los diccionarios `dicc_confl_gp` y `dicc_viol_gp` fueron extraídos de Albrieu y Palazzo (2020).

### Usage

```
data(acep_diccionarios)
```

### Format

Es un objeto de clase 'list' con 3 componentes.

**dicc\_confl\_gp** es un vector con palabras de un diccionario de terminos que refeiren a conflictos

**dicc\_viol\_gp** es un vector con palabras de un diccionario de terminos que refeiren a violencia

**dicc\_confl\_sismos** es un vector con palabras de un diccionario de terminos que refeiren a conflictos

### Source

Revista Puerto

La Nueva

### References

Albrieu, Ramiro y Gabriel Palazzo 2020 «Categorización de conflictos sociales en el ámbito de los recursos naturales: un estudio de las actividades extractivas mediante la minería de textos». Revista CEPAL (131):29-59. ([Revista CEPAL](#))

Laitano, Guillermina y Agustín Nieto «Análisis computacional de la conflictividad laboral en Mar del Plata durante el gobierno de Cambiemos». Ponencia presentado en VI Workshop - Los conflictos laborales en la Argentina del siglo XX y XXI: un abordaje interdisciplinario de conceptos, problemas y escalas de análisis, Tandil, 2021.

### Examples

```
diccionario <- acep_load_base(acep_diccionarios$dicc_viol_gp)
diccionario
```

---

acep\_frec                      *Frecuencia de palabras totales.*

---

**Description**

Funcion que cuenta la frecuencia de palabras totales en cada una de las notas/textos.

**Usage**

```
acep_frec(x)
```

**Arguments**

x                      vector de textos al que se le aplica la funcion de conteo de la frecuencia de palabras.

**Value**

Si todas las entradas son correctas, la salida sera un vector con una frecuencia de palabras.

**Examples**

```
acep_frec("El SUTEBA fue al paro. Reclaman mejoras salariales.")
```

---

acep\_int                      *Indice de intensidad.*

---

**Description**

Funcion que elabora un indice de intensidad en base a la relacion entre palabras totales y palabras del diccionario presentes en el texto.

**Usage**

```
acep_int(pc, pt, decimales = 4)
```

**Arguments**

pc                      vector numerico con la frecuencia de palabras conflictivas presentes en cada texto.

pt                      vector de palabras totales en cada texto.

decimales              cantidad de decimales, por defecto tiene 4 pero se puede modificar.

**Value**

Si todas las entradas son correctas, la salida sera un vector numerico.

**Examples**

```
conflictos <- c(1, 5, 0, 3, 7)
palabras <- c(4, 11, 12, 9, 34)
acep_int(conflictos, palabras, 3)
```

---

acep\_load\_base                    *Carga bases de datos creadas por el Observatorio.*

---

**Description**

Funcion para cargar bases de datos disponibles online. Por ahora estan disponibles las siguientes bases: Revista Puerto 'rp\_mdp'; La Nueva 'ln\_bb', La Capital 'lc\_mdp', Ecos Diarios 'ed\_neco', La Nacion 'ln\_arg'

**Usage**

```
acep_load_base(tag)
```

**Arguments**

tag                    etiqueta identificatoria del data frame a cargar: acep\_bases\$rp\_mdp, acep\_bases\$ln\_bb, acep\_bases\$lc\_mdp, acep\_bases\$ed\_neco, acep\_bases\$ln\_arg

**Value**

Si todas las entradas son correctas, la salida sera una base de datos en formato tabular con un corpus de notas.

**Examples**

```
bd_sismos <- acep_bases$rev_puerto
acep_load_base(tag = bd_sismos) |> head()
```

---

acep\_men                    *Frecuencia de menciones de palabras.*

---

**Description**

Funcion que cuenta la frecuencia de menciones de palabras que refieren a conflictos en cada una de las notas/textos.

**Usage**

```
acep_men(x, y, tolower = TRUE)
```



**Arguments**

x	vector de textos al que se le aplica la funcion de conteo de la frecuencia de menciones de palabras del diccionario.
y	vector de palabras del diccionario utilizado.
tolower	convierte los textos a minusculas.

**Value**

Si todas las entradas son correctas, la salida sera un vector con una frecuencia de palabras de un diccionario.

**Examples**

```
df <- data.frame(texto = c("El SUTEBA fue al paro. Reclaman mejoras salariales.",
"El SOIP lleva adelante un plan de lucha con paros y piquetes."))
diccionario <- c("paro", "lucha", "piquetes")
df$detect <- acep_men(df$texto, diccionario)
df
```

---

acep\_plot\_rst

*Resumen visual de la serie temporal de los indices de conflictividad.*

---

**Description**

Funcion que devuelve un panel visual de cuatro graficos de barras con variables proxy de los indices de conflictividad agrupados por segmento de tiempo.

**Usage**

```
acep_plot_rst(datos, tagx = "horizontal")
```

**Arguments**

datos	data frame con datos procesados.
tagx	orientacion de las etiquetas del eje x ('horizontal'   'vertical').

**Value**

Si todas las entradas son correctas, la salida sera una imagen de cuatro paneles.

**Examples**

```
datos <- acep_bases$rp_procesada
fecha <- datos$fecha
n_palabras <- datos$n_palabras
conflictos <- datos$conflictos
datos_procesados_anio <- acep_rst(datos,
fecha, n_palabras, conflictos, st = 'anio')
acep_plot_rst(datos_procesados_anio, tagx = 'vertical')
```

---

`acep_plot_st`*Grafico de barras de la serie temporal de indices de conflictividad.*

---

**Description**

Funcion que devuelve un grafico de barras con la serie temporal de indices de conflictividad por dia, mes o anio.

**Usage**

```
acep_plot_st(x, y, t = "", ejex = "", ejey = "", etiquetax = "horizontal")
```

**Arguments**

<code>x</code>	vector de valores del eje x (por ejemplo, fechas).
<code>y</code>	vector de valores numericos del eje y (por ejemplo, menciones).
<code>t</code>	titulo del grafico.
<code>ejex</code>	nombre del eje x.
<code>ejey</code>	nombre del eje y.
<code>etiquetax</code>	orientacion de las etiquetas del eje x ('horizontal'   'vertical').

**Value**

Si todas las entradas son correctas, la salida sera una imagen de un panel.

**Examples**

```
datos <- acep_bases$rp_procesada
fecha <- datos$fecha
n_palabras <- datos$n_palabras
conflictos <- datos$conflictos
dpa <- acep_rst(datos,
fecha, n_palabras, conflictos, st = 'anio')
acep_plot_st(
dpa$st, dpa$freem,
t = 'Evolucion de la conflictividad en el sector pesquero argentino',
ejex = 'Anios analizados',
ejey = 'Menciones de terminos del diccionario de conflictos',
etiquetax = 'horizontal')
```

---

`acep_rs`*Cadenas de caracteres para limpiar y normalizar textos.*

---

**Description**

Cadenas de caracteres y expresiones regulares para limpiar y normalizar textos.

**Usage**

```
data(acep_rs)
```

**Format**

Son cadenas de caracteres.

**stopwords** es un string de palabras vacias.

**dias** es un string de dias.

**meses** es un string de meses.

**emoji** es un string con expresiones regulares para emojis.

**sintildes** es un string de letras sin tildes.

**tildes** es un string de letras con tildes.

**Examples**

```
print(acep_rs)
```

---

`acep_rst`*Serie temporal de indices de conflictividad.*

---

**Description**

Funcion que devuelve los indices de conflictividad agrupados por segmento de tiempo: dia, mes, anio.

**Usage**

```
acep_rst(datos, fecha, frecp, frecm, st = "mes", u = 2, d = 4)
```

**Arguments**

datos	data frame con los textos a procesar.
fecha	columna de data frame que contiene el vector de fechas en formato date.
frecp	columna de data frame que contiene el vector de frecuencia de palabras por texto.
frecm	columna de data frame que contiene el vector de menciones del diccionario por texto.
st	parametro para establecer el segmento temporal a ser agrupado: anio, mes, dia.
u	umbral de menciones para contabilizar una nota como nota que refiere a un conflicto.
d	cantidad de decimales, por defecto tiene 4 pero se puede modificar.

**Value**

Si todas las entradas son correctas, la salida sera una base de datos en formato tabular con nuevas variables.

**Examples**

```
datos <- acep_bases$rp_procesada
fecha <- datos$fecha
n_palabras <- datos$n_palabras
conflictos <- datos$conflictos
datos_procesados_anio <- acep_rst(datos,
  fecha, n_palabras, conflictos, st = 'anio', u = 4)
datos_procesados_mes <- acep_rst(datos,
  fecha, n_palabras, conflictos)
datos_procesados_dia <- acep_rst(datos,
  fecha, n_palabras, conflictos, st = 'dia', d = 3)
datos_procesados_anio |> head()
datos_procesados_mes |> head()
datos_procesados_dia |> head()
```

---

acep\_token

*Tokenizador.*

---

**Description**

Funcion que tokeniza las notas/textos.

**Usage**

```
acep_token(x, tolower = TRUE)
```

**Arguments**

x                   vector de textos al que se le aplica la funcion de tokenizacion.  
tolower            convierte los textos a minusculas.

**Examples**

```
acep_token("Huelga de obreros del pescado en el puerto")
```

---

acep\_token\_plot       *Grafico de barras de palabras mas recurrentes en un corpus.*

---

**Description**

Funcion que devuelve un grafico de barras con las palabras mas recurrentes en un corpus textual.

**Usage**

```
acep_token_plot(x, u = 10, frec = TRUE)
```

**Arguments**

x                   vector de palabras tokenizadas.  
u                   numero de corte para el top de palabras mas frecuentes.  
frec                parametro para determinar si los valores se visualizaran como frecuencia absoluta o relativa.

**Value**

Si todas las entradas son correctas, la salida sera un grafico de barras.

**Examples**

```
tokens <- c(rep("paro",15), rep("piquete",25), rep("corte",20), rep("manifestacion",10),  
rep("bloqueo",5), rep("alerta",16), rep("ciudad",12), rep("sindicato",11), rep("paritaria",14),  
rep("huelga",14), rep("escrache",15))  
acep_token_plot(tokens)
```

---

acep_token_table	<i>Tabla de frecuencia de palabras tokenizadas.</i>
------------------	---

---

**Description**

Funcion que cuenta la frecuencia de palabras tokenizadas.

**Usage**

```
acep_token_table(x, u = 10)
```

**Arguments**

x	vector de palabras tokenizadas.
u	numero de corte para el top de palabras mas frecuentes.

**Value**

Si todas las entradas son correctas, la salida sera una tabla con la frecuencia relativa y absoluta de palabras tokenizadas.

**Examples**

```
tokens <- c(rep("paro",15), rep("piquete",25), rep("corte",20), rep("manifestacion",10),  
rep("bloqueo",5), rep("alerta",16), rep("ciudad",12), rep("sindicato",11), rep("paritaria",14),  
rep("huelga",14), rep("escrache",15))  
acep_token_table(tokens)
```

# Index

- \* **datos**
    - acep\_bases, 2
    - acep\_load\_base, 8
  - \* **diccionarios**
    - acep\_diccionarios, 6
  - \* **expresiones**
    - acep\_rs, 11
  - \* **indicadores**
    - acep\_db, 4
    - acep\_detect, 5
    - acep\_frec, 7
    - acep\_int, 7
    - acep\_men, 8
  - \* **normalizacion**
    - acep\_clean, 3
  - \* **regulares**
    - acep\_rs, 11
  - \* **resumen**
    - acep\_rst, 11
  - \* **tablas**
    - acep\_token\_table, 14
  - \* **tokenizar**
    - acep\_token, 12
  - \* **visualizacion**
    - acep\_plot\_rst, 9
    - acep\_plot\_st, 10
    - acep\_token\_plot, 13
- 
- acep\_bases, 2
  - acep\_clean, 3
  - acep\_db, 4
  - acep\_detect, 5
  - acep\_diccionarios, 6
  - acep\_frec, 7
  - acep\_int, 7
  - acep\_load\_base, 8
  - acep\_men, 8
  - acep\_plot\_rst, 9
  - acep\_plot\_st, 10
  - acep\_rs, 11
  - acep\_rst, 11
  - acep\_token, 12
  - acep\_token\_plot, 13
  - acep\_token\_table, 14