mincaqdasr. Herramienta minimalista para anotado de documentos

Autores:

## Justificación

mincaqdasr es un herramienta para análisis cualitativo (CAQDAS), particularmente para la tarea de codificar o anotar un corpus de documentos o fragmentos de los mismos, a través de una interfaz gráfica (GUI) que corre en cualquier navegador que tenga JavaScript habilitado.

En investigación cualitativa, la codificación es un proceso clave (Saldaña, 2016). Otro ejemplo importante se encuentra en (Auerbach & Silverstein, 2003).

En investigación cualitativa una tarea esencial es la de “codificar”: “A code in qualitative inquiry is most often a word or short phrase that symbolically assigns a summative, salient, essence-capturing, and/or evocative attribute for a portion of language-based or visual data”1. El objetivo de mincaqdasr es ofrecer una interfaz gráfica sencilla para facilitar este proceso, que puedar ser incluida en el código de un análisis mayor (donde, por ejemplo, se genera el corpus de documentos, o se procesen correlaciones entre los códigos asignados), y que ayude a volver nuestro análisis más transparente y transportable.

El corpus anotado se exporta en un archivo .json que contiene todos los elementos de un proyecto, de modo modo que no es necesario hacer bundles, carpetas, ni exportar nada más.

## Estructura de datos

Los datos se guardan en .json que incluye 4 elementos primarios: \* documentos (como un vector de texto); \* códigos (un vector de texto) que se aplican tanto para documentos como a fragmentos; \* anotaciones a nivel documento, que incluyen la referencia a uno o varios códigos y un memo del anotador; \* fragmentos y anotaciones a nivel fragmentos, que incluyen la referencia al documento que contiene el fragment, la referencia a uno o varios códigos, y memo del anotador.

Un ejemplo de este .json corresponde a un análisis sobre sentidos de la partenidad (The Yeshiva University Fatherhood Project) que Auerbach y Silverstein incluyen en su libro sobre investigación cualitativa, codificación y grounded theory2, y que los autores analizan paso a paso: la lectura de los documentos, selección de los fragmentos, anotación de comentarios del analista, y generación y asignación de códigos o etiquetas. Todos estos pasos se pueden realizar con mincaqdasr.

En R, estos .json pueden ser importado en R con jsonlite::read\_json() y manipulados como una lista. El package de mincaqsdar incluye algunas funciones para simplificar esta manipulación de datos.

Otro ejemplo de .json es sobre fragmentos de noticias de la prensa argentina acerca de “big data”. En este caso no se asignan códigos a fragmentos, sino que cada documento es clasificado con una etiqueta de “positivo” o “negativo”, de acuerdo a si el big data queda asociado a una potencial aplicación beneficiosa o a un riesgo. El ejemplo se utiliza en este tutorial3, donde se cita el corpus más amplio.

## Usar mincaqdasr en R

Cuando llamamos a mincaqdasr::start\_gui() estamos abriendo un .html adentro de una shinyapp (no es la mejor implementación, pero es rápida; lo mejor sería abrir el navegador sin llamar una shinyapp). Esto significa que R queda “escuchando” la shinyapp de modo que vamos a ver en nuestra consola un mensaje tipo Listening on <http://127.0.0.1:7654>. Una vez que hayamos terminado de trabajar con el GUI debemos cerrar este proceso para poder seguir usando R. You can stop the app and return access to the console using any one of these options: Click the stop sign icon on the R console toolbar. Click on the console, then press Esc (or press Ctrl + C if you’re not using RStudio). Close the Shiny app window. Más info sobre esto acá: <https://mastering-shiny.org/basic-app.html#running> Usar mincaqdasr sin R El GUI está pensado para correr de manera autónoma, de modo que se puede utilizar sin llamarlo desde R, ni instalar todo el package. En cuyo caso, sólo hay que descargar la carpeta que tiene 3 archivos llamados “gui” (.html, .js, .css) y 2 .json de ejemplo. Para correr el GUI hay que abrir el archivo inst/gui.html con un navegador que tenga JavaScript habilitado. Los documentos y códigos de anotación se pueden incluir directamente dentro del <script></script>, en una variable llamada input\_data. En el .html ya hay un ejemplo comentado que se puede editar.

## Otras herramientas

* Simil NVivo o Atlas.ti pero minimalista, de modo que se incluye más fácilmente en tu propio *workflow* de trabajo y otras tareas del análisis (scrapeo, limpieza, etc.)
* qcoder requiere armar carpeta
* rqda requeire GFTk2 discontinuado

# Referencias

Auerbach, C. F., & Silverstein, L. B. (2003). Qualitative data: An introduction to coding and analysis. *NYU Press*.

Saldaña, J. (2016). *The coding manual for qualitative researchers*. SAGE Publications.