ÍNDICE

Índice de imágenes, tablas y gráficos de los Anexos

ANEXOS

I. Tablas de abreviaturas y definiciones

Tabla de abreviaturas

API	*Application Programming Interface*	**RRSS**	Redes Sociales
CAP ó PAC	Política Agraria Común	**RT**	*Retuit*
DL	*Deep Learning*	**SA**	*Sentiment analysis*
EM	Estados Miembros (de la UE)	**SFSC**	*Short Food Supply Chain*
GMOs ó OGM	Organismos Genéticamente Modificados	**SMA**	*Social Media Analytics*
IA	Inteligencia Artificial	**SOC**	*Soil Organic Carbon*
IC	Inferencia Causal	**TDT**	*Topic Detection and Tracking*
MAPA	Ministerio de Agricultura, Pesca y Alimentación	**TF-IDF**	*Term Frequency-Inverse Document Frequency*
ML	*Machine Learning*	**TIC**	Tecnologías de la Información y la Comunicación
NLP	Procesamiento del Lenguaje Natural	**UE**	Unión Europea
ODS	Objetivos de Desarrollo Sostenible	**UGC**	*User-Generated Content*
PEPAC	Plan Estratégico de la PAC de España	**URL**	*Uniform Resource Locator*

Tabla de definiciones

API	Interfaz que comunica dos aplicaciones o *software* pertenecientes a organizaciones distintas o bien que cumplen funciones distintas, estableciendo protocolos de comunicación entre sí mediante solicitudes y respuestas.
Benchmarking	En ML, el *benchmarking* es la práctica de comparar herramientas para identificar los algoritmos que ofrecen un mejor desempeño. Se trata de una tarea compleja debido a la gran cantidad de factores que intervienen en el rendimiento de los modelos ML.
Corpus	Un corpus es una colección de texto escrito por un nativo y organizado en *datasets* para entrenar sistemas de inteligencia artificial y aprendizaje automático NLP. Un corpus puede estar compuesto de múltiples tipologías de texto, desde periódicos, novelas o recetas hasta *tui*
Dataset	Conjuntos de datos que presentan una estructura contenida en una única, donde cada columna representa una variable en particular, y cada fila representa a un miembro determinado del conjunto de datos a tratar.
Dataframe	Se trata de un *dataset* donde cada columna presenta un nombre y además admiten valores alfanuméricos.
F-Score	En estadística, el valor F es la media armónica que combina los valores de precisión y exhaustividad (*recall*). Se utiliza para poner a prueba algoritmos de búsqueda y recuperación de información y clasificación de documentos.

Hashtag	Un *hashtag* o almohadilla es una palabra o grupo de palabras clave precedidas por una almohadilla (#) que identifican un tema y sobre las que un usuario en RRSS puede clicar para visualizar otras publicaciones relacionadas con ese mismo tema. Sirven de etiqueta con el fin de que tanto el sistema como el usuario la identifiquen de forma rápida y extiendan su uso.
Lematización ó *stemming*	Proceso utilizado en el NLP que consiste en reducir el tamaño y complejidad de un texto mediante la agrupación de todas las palabras flexionadas o derivadas de un mismo término en su forma canónica o lema. Por ejemplo, el lema de 'niña', 'niño', 'niñita', 'niños' es el vocablo 'niño'.
Lexicon	Serie ordenada de palabras de una misma lengua, materia o época determinadas.
Stop words	Las *stop words* (o palabras vacías) son un término nacido en el lenguaje informático que hace referencia a aquellas palabras que no agregan mucha información al texto sino que cumplen un papel funcional (como artículos, preposiciones, pronombres, conjunciones, adverbios, etc.). Normalmente, las técnicas de procesamiento del lenguaje excluyen estas palabras para eliminar la información de bajo nivel y de esta forma reducir el tamaño del conjunto de datos.
TF-IDF	Modelo matemático empleado en el NLP para estimar la relevancia de un documento para un término, cuyo funcionamiento está basado en dos premisas. Lo que hace es medir la frecuencia con la que aparece un término o frase dentro de un documento determinado (*term frequency*), y lo compara con el número de documentos que mencionan ese término dentro de una colección entera de documentos (*inverse document frequency*).
Token	Un *token* es una una unidad semántica independiente dentro de una secuencia de un texto. Puede tratarse de palabras, caracteres especiales, símbolos u oraciones que, agrupados, forman una unidad de información útil en el procesamiento de esos datos.
Tuit* ó *Tweet	Unidad de texto escrito en la red social Twitter, con una extensión máxima de280 caracteres por publicación.
User-Generated Content	Todo contenido creado por los usuarios de Internet a través de las redes sociales o páginas web, en diferentes formatos: publicaciones en blogs, fotos, vídeos, *tuits*, reseñas, etc.

II. Metodología

– *Definición de los requerimientos y determinación de la muestra*

Los datos se recogerán y analizarán utilizando el software R, uno de los lenguajes de programación más empleados para computación estadística y gráfica, analítica de datos así como para técnicas NLP. Se trata de un código abierto, gratuito y muy completo; cuenta con miles de paquetes o librerías (colecciones de funciones y conjunto de datos desarrollados por la comunidad usuaria) a libre disposición para que otros puedan utilizarlos en sus propios proyectos y optimizarlos, en un proceso de mejora continua del código abierto. Se trata, además, de un lenguaje relativamente sencillo de aprender, por lo que es una buena opción para aquellos investigadores que no hayan tenido un contacto previo con lenguajes de programación.

A través de sus paquetes (como ***rtweet***, ***ROAuth*** y ***httpuv***), proporciona comunicación con la interfaz de programación de aplicaciones (API por sus siglas en inglés) de Twitter, permitiendo la búsqueda de *tuits* por medio de palabras clave específicas y posteriormente, su extracción y almacenamiento en una base de datos. Este paquete extrae todos aquellos *tuits* que contengan la palabra clave introducida para un periodo de tiempo determinado. Para dicha extracción, el programador también puede especificar el idioma en el que han de estar escritos los *tuits*, así como la ubicación geográfica específica desde la que se publicaron, por lo que cualquier *tuit*.

¿Por qué Twitter? Gracias a la API de Twitter, cuyo acceso está abierto a cualquier usuario que tenga una cuenta de desarrollador, Twitter ha supuesto una fuente de datos importante para la investigación académica en numerosos campos (Antypas, Preece & Collados, 2022), incluidas las técnicas NLP. De esta forma, se ha convertido en una de las RRSS, junto a Reddit, que más facilita la extracción y el tratamiento de los datos publicados por sus usuarios. Teniendo en cuenta el alcance de esta plataforma en cuanto a número de usuario y el peso de la información política dentro de la misma (según una investigación de 2019 comisionada por Twitter España, el 64% de sus usuarios utilizan Twitter al menos una vez a la semana para informarse de temas políticos), se trata de la red social que más datos podrá proporcionar para el análisis de sentimientos de este trabajo.

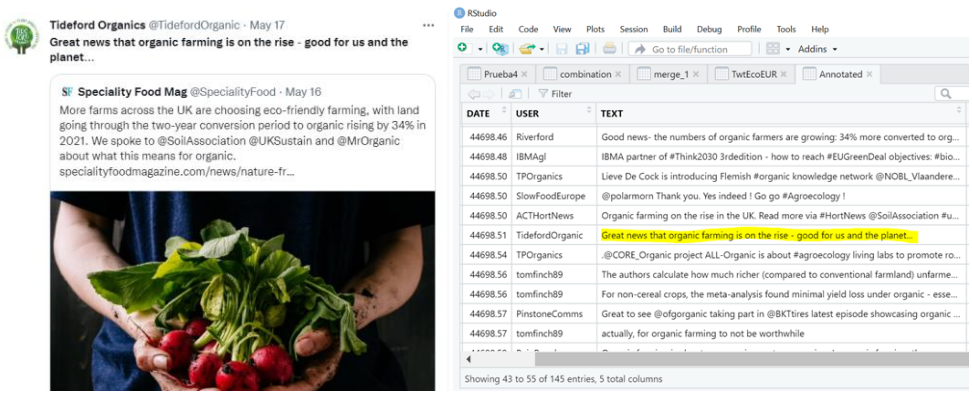


Imagen 1. *Un mismo* tuit *visto desde la aplicación de Twitter y desde RStudio.*
Fuente: Twitter y RStudio.

Las API son un conjunto de definiciones y protocolos que se utilizan en informática para integrar el software de una aplicación en otra. Las API permiten que los productos y servicios de un medio se comuniquen con otros, sin necesidad de saber cómo están

implementados. Esto simplifica el desarrollo de las aplicaciones y permite a los programadores evitar el trabajo de programar la implementación con el software desde cero. Para el caso concreto de la API de Twitter, Twitter permite obtener de forma gratuita datos de los perfiles de los usuarios y sus publicaciones en esta red (*tuits*, espacios, listas, usuarios y más) para así analizarlos, e incluso crear una aplicación propia. Para poder obtener datos de Twitter a través de su API, es necesario registrarse en su plataforma y crear una App de Twitter asociada (**ver Anexo III**). Twitter App es el mecanismo que proporciona Twitter para desarrolladores que quieran acceder a los contenidos de Twitter a través de otros programas, como RStudio. Al crear una Twitter App, Twitter proporciona una serie de *tokens* y claves de identificación para poder acceder a la aplicación y desde ahí extraer información.

La última versión, Twitter API v2, ofrece distintos niveles de acceso a los usuarios en función de los requerimientos que presente su proyecto:

	Essential	Elevated	Academic Research
Precio	Gratuito	Gratuito	Gratuito
Acceso a versiones anteriores de la API	Denegado	Permitido	Permitido
Límite de Apps	1	3	1
Límite de extracción de datos	500.000 tweets/mes	2M tweets/mes	10M tweets/mes
Filtrado de datos	Máx. 5 reglas/flujo	Máx. 25 reglas/flujo	Máx. 1.000 reglas/flujo
Período temporal de *tuits* extraídos	Últimos 9 días	Últimos 9 días	Ilimitado

Tabla 1. *Características de los accesos a la API Twitter.*
Fuente: Elaboración propia a partir del Twitter Developer Portal.

Para el presente trabajo, se ha solicitado el acceso "Academic Research" a través del Developer Portal de Twitter por varias razones. En primer lugar, permite acceder a versiones anteriores de la API. Esto es necesario para poder trabajar con algunas de las librerías de R que hemos seleccionado, las cuales habían sido probadas con la versión anterior de la API. En segundo lugar, el acceso "Academic Research" permite al usuario introducir filtros en la búsqueda de *tuits* (hasta 1.000 condiciones), de forma que devolverá un *input* más exacto (cuantos más condicionantes se incluyan) y facilitará posteriormente la "limpieza" de *tuits*. Por último, un aspecto restrictivo de los accesos "Essential" y "Elevated" es el límite temporal de extracción de *tuits* a los últimos 9 días. Esto impide poder recoger grandes volúmenes de datos y llevar a cabo una comparativa sobre la evolución del sentimiento de la ciudadanía a lo largo de varios años. El acceso Academic Research permite la búsqueda de *tuits* sin restricciones en el periodo temporal, permitiendo hacer un diagnóstico más completo sobre cualquier materia.

Los datos: los tuits.

Para el presente trabajo, es necesario conseguir una muestra que sea lo más representativa posible del conjunto de la sociedad. Para ello, hay que tener en cuenta diversos aspectos que afectarán a la muestra final que se consiga extraer, así como a los futuros resultados que se obtengan. Las características que se deben tener presentes son:

1. **Volumen de datos:** el volumen de la muestra de *tuits* depende en gran medida del tipo de acceso que se tiene a la API de Twitter, el idioma de los *tuits* o bien el rango temporal con el que se trabaje. Diferentes estudios han tratado de determinar cuál es el tamaño mínimo necesario para que una muestra de datos pueda emplearse en técnicas NLP, así como el límite por encima del cual agregar más datos no presenta ningún valor añadido. En Olthof, A. W. et al (2021), la especificidad y el valor predictivo de los modelos DL de procesamiento del lenguaje que desarrollaron para la extracción de información útil de informes de radiología alcanzaron su máximo valor alrededor de las 800-1.000 informes y se estancaron después de eso. Por otro lado, Prusa & Seliya (2015) trabajaron con *datasets* de *tuits* de distintos tamaños (desde 1.000 hasta 297.000 *tuits*) para entrenar modelos ML de clasificación de sentimientos y concluyeron que los modelos aumentaban su rendimiento con el tamaño de la muestra. Sin embargo, esta mejora dejaba de ser significativa a partir de muestras de más de 81.000 *tuits*. En el caso de los métodos de aprendizaje no supervisado (ver Apartado 2.3.), el tamaño del *lexicon* (la cantidad de palabras negativas y positivas) también influye en la precisión de los resultados, al tratarse de la base de datos frente a la que se contrasta el conjunto de datos en origen (Mitra, 2020).
2. **Zona geográfica:** conocer la localización de los usuarios que *tuitean* puede resultar muy útil en el análisis del UGC. Entre otras cosas, permite conocer las diferencias existentes entre las posturas de distintos países, así como asociar una opinión que es mayoritaria dentro de una región particular a unas condiciones geográficas, socioculturales y económicas concretas (incluso climáticas). El paquete de R ***rtweet*** tiene la opción de extraer *tuits* con una geolocalización concreta. Para ello, el usuario puede introducir las coordenadas y seleccionar el radio de interés, o bien introducir el nombre del país del que desea conocer esa información. No obstante, introducir este filtro presenta una desventaja ya que supone eliminar una gran cantidad de *tuits* de la muestra, como se verá más adelante (ver subapartado 4.1. Caso 1. Agricultura ecológica), puesto que con esta selección solo se pueden recoger aquellos *tuits* de usuarios que, en el momento de publicar el *tuit*, tenían activada la geolocalización de Twitter.
3. **Idioma:** las técnicas de *opinion mining* son más precisas cuantos más datos puedan procesar. El idioma más utilizado en Twitter es el inglés. El 34% de todos los *tuits* están escritos en este idioma (Statista, 2013). Asimismo, los *lexicon*, *benchmark* y modelos ML disponibles para tareas de analítica de sentimientos están mayoritariamente desarrollados en inglés, limitando la investigación en otras lenguas (Rodríguez-Ibáñez, M. et al, 2021). Son muchas las investigaciones en las que el conjunto de datos ha sido traducido desde un idioma al inglés para poder trabajar con herramientas ya existentes. También los *lexicon* disponibles en inglés han sido traducidos a otras lenguas, dando lugar a imprecisiones derivadas del proceso de traducción que no tiene en cuenta las particularidades del segundo idioma (Rodríguez-Ibáñez, M. et al, 2021). Por tanto, el inglés es el idioma que dispone de un mayor número de datos y recursos para las técnicas NLP. No obstante, la selección de un idioma u otro dependerá en última instancia de los requerimientos o la temática del estudio en cuestión.
4. **Periodo temporal:** conocer la evolución de las opiniones de los ciudadanos a lo largo del tiempo permite detectar futuras tendencias, identificar episodios

disruptivos (como el movimiento #MeToo dentro del feminismo) y actuar a tiempo cuando el interés por un tema está desapareciendo o bien el sentimiento generalizado hacia mismo empieza a ser muy negativo. Se trata de uno de los análisis más esclarecedores que pueden hacerse en el campo del análisis de sentimientos.

5. **Modo de extracción:** para el análisis de UGC en RRSS, lo más habitual es recoger la información de manera asíncrona, es decir, descargando o extrayendo publicaciones y archivos (como *tuits*) de cualquier fecha anterior para su análisis (Salmons, 2017). Los miembros pueden regresar y revisar los materiales publicados en el pasado para leerlos.
6. **Palabras clave:** la selección de aquellas palabras en las que se basará la búsqueda de *tuits* es otro punto crítico. Debe procurarse que estas no sean palabras muy genéricas, ni tampoco demasiado excluyentes, y de que los usuarios las utilicen para referirse en exclusiva a la temática de interés (Borrero & Zabalo, 2021). No se recomienda el empleo de siglas pues con frecuencia hacen referencia a más de un término. En cuanto al uso de *hashtags* en Twitter como palabras clave, hay diversidad de opiniones: Borrero & Zabalo (2021), por ejemplo, apoyan el uso de *hashtags* como palabras clave al tratarse de palabras que el usuario ha querido enfatizar de forma intencionada. Sin embargo, el uso de la almohadilla puede dejar fuera de la muestra a todos aquellos *tuits* donde el usuario haya hablado del tema sin haber introducido el *hashtag*, arriesgándose a perder una gran cantidad de datos.

– *Pre-procesamiento de datos*

El pre-procesamiento (o limpieza) de textos, dentro del ámbito NLP, consiste en eliminar del texto todos aquellos elementos que no aporten información sobre su temática, contenido o estructura. El proceso de limpieza de un texto es el primer y más importante paso en cualquier proceso de minería de datos. Implica convertir datos sin procesar en un formato que los algoritmos NLP puedan interpretar. No existe una sola forma de hacerlo, sino que dependerá de la finalidad del análisis y de la fuente de la que proceda el texto. Esto ayudará a la obtención de unos resultados más ajustados a la realidad.

Pueden incluirse toda una serie de tareas como la retirada de *retuits* (RT) para evitar la repetición de *tuits* dentro de la muestra, la eliminación de *links* y emoticonos (*emojis*), la conversión del texto a minúsculas o la lematización. Esta operación puede extenderse tanto como el programador quiera, en función de las necesidades del estudio (¿es preferible trabajar con una muestra a pequeña escala, procesada lo máximo posible para mejorar la precisión del modelo NLP? ¿o bien es mejor trabajar con una muestra grande de datos más "sucios" para testear el modelo en una situación "real", logrando con ello ahorrar tiempo y esfuerzo computacional?).

A continuación, el Gráfico 1 muestra paso a paso el pre-procesamiento de un conjunto de datos procedente de RRSS.

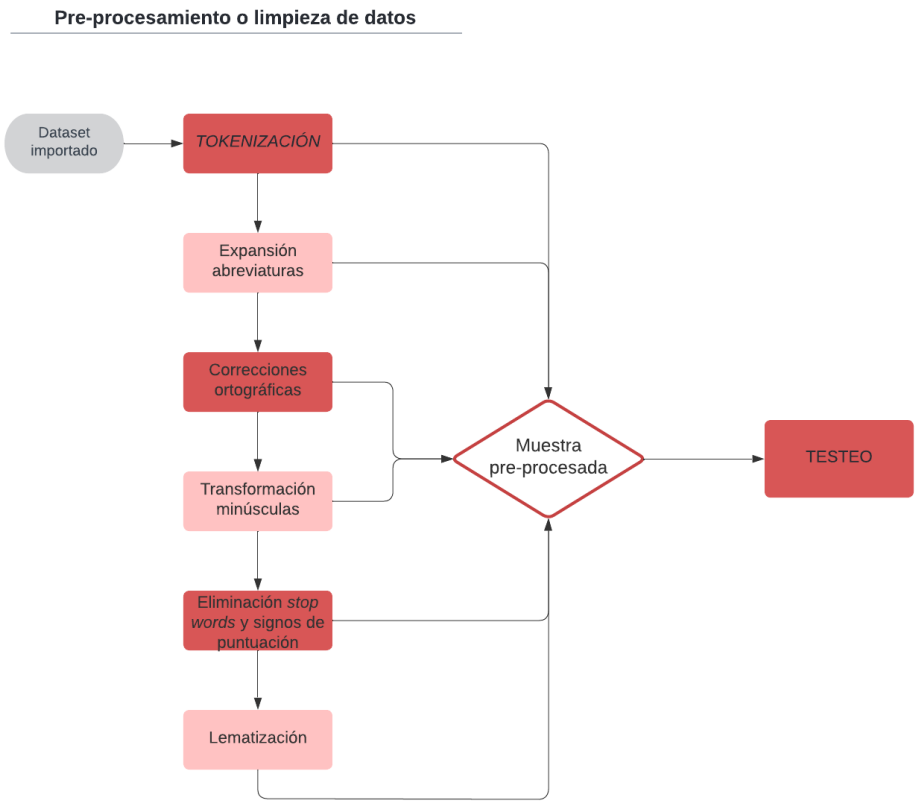


Gráfico 1. *Diagrama de flujo del pre-procesamiento de un conjunto de* tuits*.*
Fuente: Elaboración propia.

1. *Tokenización*: es el método de dividir un flujo de texto en *tokens*, entendiendo un *token* como la unidad más sencilla con significado propio dentro del texto (que pueden ser palabras, caracteres especiales, símbolos u oraciones) y eliminar caracteres especiales que no aporten información relevante para el análisis de sentimientos (Bermeo-Almeida et al, 2019). De esta forma, excluimos nombres de usuarios, *hashtags* (#DeLaGranjaALaMesa), URLs (https://), emoticonos, etc.

2. Expansión de las abreviaturas y notaciones abreviadas para que los algoritmos NLP puedan reconocerlas.

3. Corrección ortográfica. Existen correctores ortográficos que reescriben palabras mal escritas contenidas en el texto.

4. Transformación del texto a minúsculas: el uso de mayúsculas es poco frecuente en el lenguaje de programación. Para evitar problemas en la interpretación del *dataset*, es recomendable trabajar siempre con minúsculas.

5. Eliminación de signos de puntuación (“¿”, “; ”, “…”) así como de *stop words* (artículos, conjunciones, preposiciones, adverbios, etc.). De esta forma, se simplifica la muestra eliminando aquellas palabras y símbolos que no ofrecen información sobre el sentimiento del texto, facilitando el procesamiento del conjunto de datos debido a la menor cantidad de *tokens* involucrados en el entrenamiento.

6. Lematización: Se utiliza para agrupar formas flexionadas de una misma palabra, dejando solo la raíz o lema. Ejemplo: la raíz o de los verbos “estudié”, “estudiabas”,

“estudiaremos” es 'estudiar'. Esto reducirá la cantidad de palabras de un *tuit* y por tanto su complejidad (Susmitha&Pranitha, 2022).

- *Testeo*

En el Subapartado 2.3.2.4. Comparativa de los distintos métodos para el análisis de sentimientos, se ha hablado de cómo la elección de un método de aprendizaje automatizado sobre otro depende enteramente de las condiciones particulares que presenta cada caso de estudio, así como del uso que se vaya a hacer de los mismos.

Si bien el método de aprendizaje supervisado presenta un mayor rendimiento, el método de aprendizaje *lexicon-based* permite analizar enormes cantidades de datos en poco tiempo (puesto que no requiere de una etapa de entrenamiento del modelo), obteniendo precisiones más bajas que los modelos ML pero aun así aceptables si lo que se pretende es únicamente evaluar la eficacia de los métodos de extracción y análisis automatizado de datos para la identificación, dentro del ámbito de las RRSS, de tendencias, corrientes de pensamiento, preferencias, etc., de la sociedad, tomándolos como complemento a otras herramientas de evaluación que permitan recabar datos de manera fiable.

Debido a ello, esta primera aproximación al NLP aplicado al análisis de políticas agroalimentarias se hará empleando el método de aprendizaje no supervisado. A lo largo de los dos casos de estudio a efectuar, se determinará la efectividad de distintos *lexicon* para la detección de polaridad y asimismo se hará una comparativa de los resultados frente a un análisis llevado a cabo mediante anotación manual. Este enfoque híbrido consistirá en una primera validación manual que permita seleccionar el *lexicon* más ajustado al *dataframe* (aquel que mejor se adapta al formato *tuit* de la muestra), y posteriormente continuar con el método de aprendizaje no supervisado, que permitirá automatizar el análisis de la muestra y así ahorrar tiempo. De esta forma, se garantiza la obtención de unos buenos resultados, con un balance satisfactorio entre el tiempo empleado y la precisión alcanzada.

El algoritmo de *opinion mining* utilizado en este caso (ver Apartado 4) permite analizar la polaridad a nivel de *tuit*, es decir, devuelve un único resultado global para cada *tuit* (“positivo”, “negativo”, “neutral”) en base al número de palabras de sentimiento positivo o negativo que detecte el algoritmo. Para ello, tiene en cuenta el sentimiento agregado de cada palabra y cada oración dentro del *tuit*. Esto supone una limitación en aquellos *tuits* que contienen múltiples opiniones sobre un mismo tema, puesto que el algoritmo detectará palabras que denotan sentimientos opuestos dentro de un mismo *tuit*. A lo largo de los dos casos de estudio se abordarán distintas estrategias para tratar de paliar este inconveniente y otros como la especificidad que presentan los *lexicon* dentro de un dominio específico y su limitación a la hora de detectar aspectos del lenguaje como la negación o el sarcamo (ver Apartado 5).

III. Registro y acceso a la API de Twitter

1. Es necesario tener un perfil de desarrollador en Twitter. Para ello, hay que entrar al Portal del Desarrollador de Twitter con una cuenta propia de Twitter ya iniciada y solicitar el acceso "Essential". Acto seguido, el usuario debe verificar su nombre de usuario y su correo electrónico y añadir ciertos datos personales (ver Imagen 2).

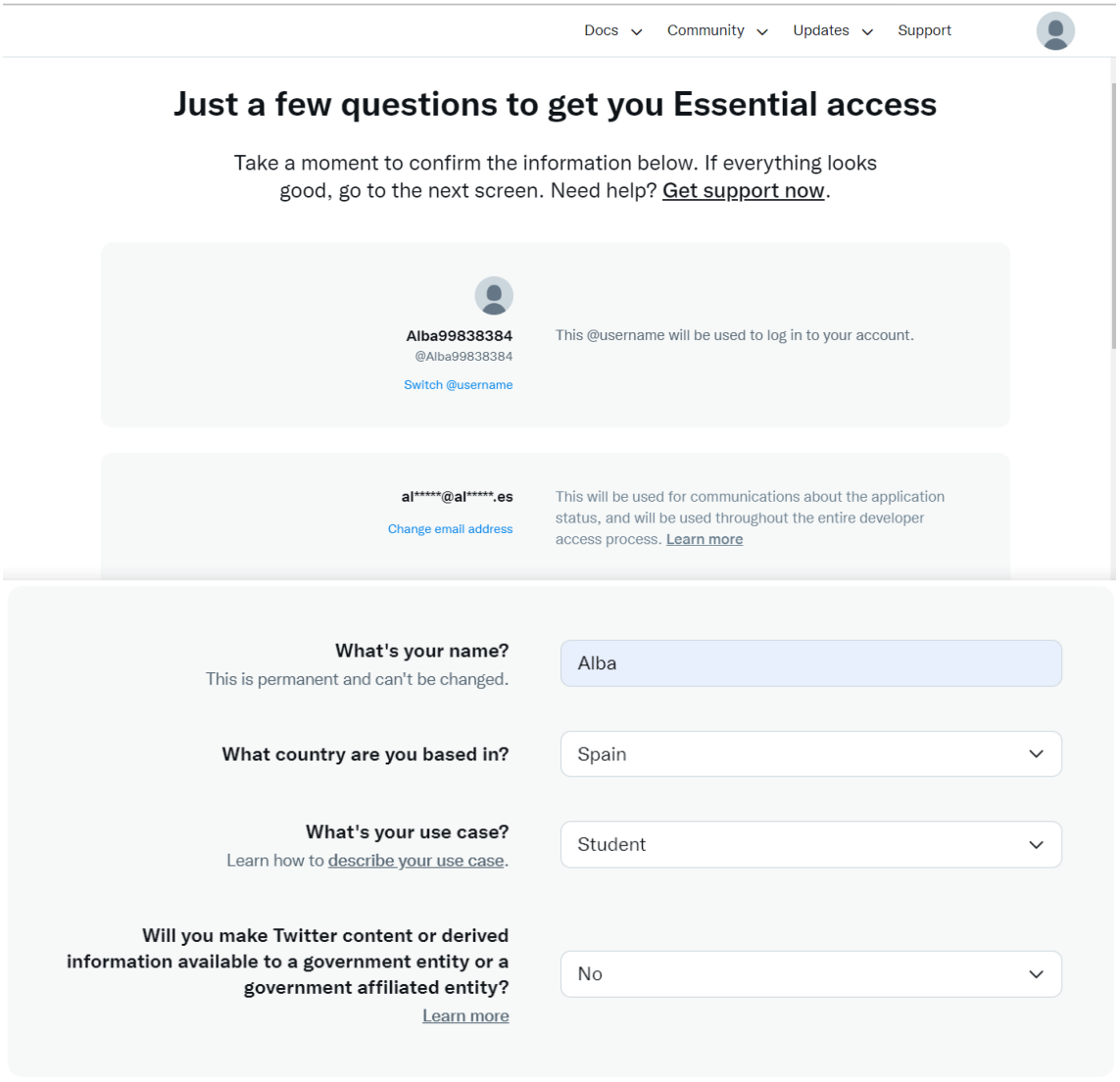


Imagen 2. *Detalle del registro al Twitter Developer Portal.*
Fuente: Twitter Developer Portal.

2. Una vez creada la cuenta de desarrollador e iniciada la sesión en el Developer Portal de Twitter, el usuario debe crear una App propia. Al hacer click en "Create an App", aparecerán las siguientes opciones para seleccionar/rellenar:

- Entorno de la App (App environment). Para este trabajo, se selecciona la opción "Desarrollo".
- Nombre de la App

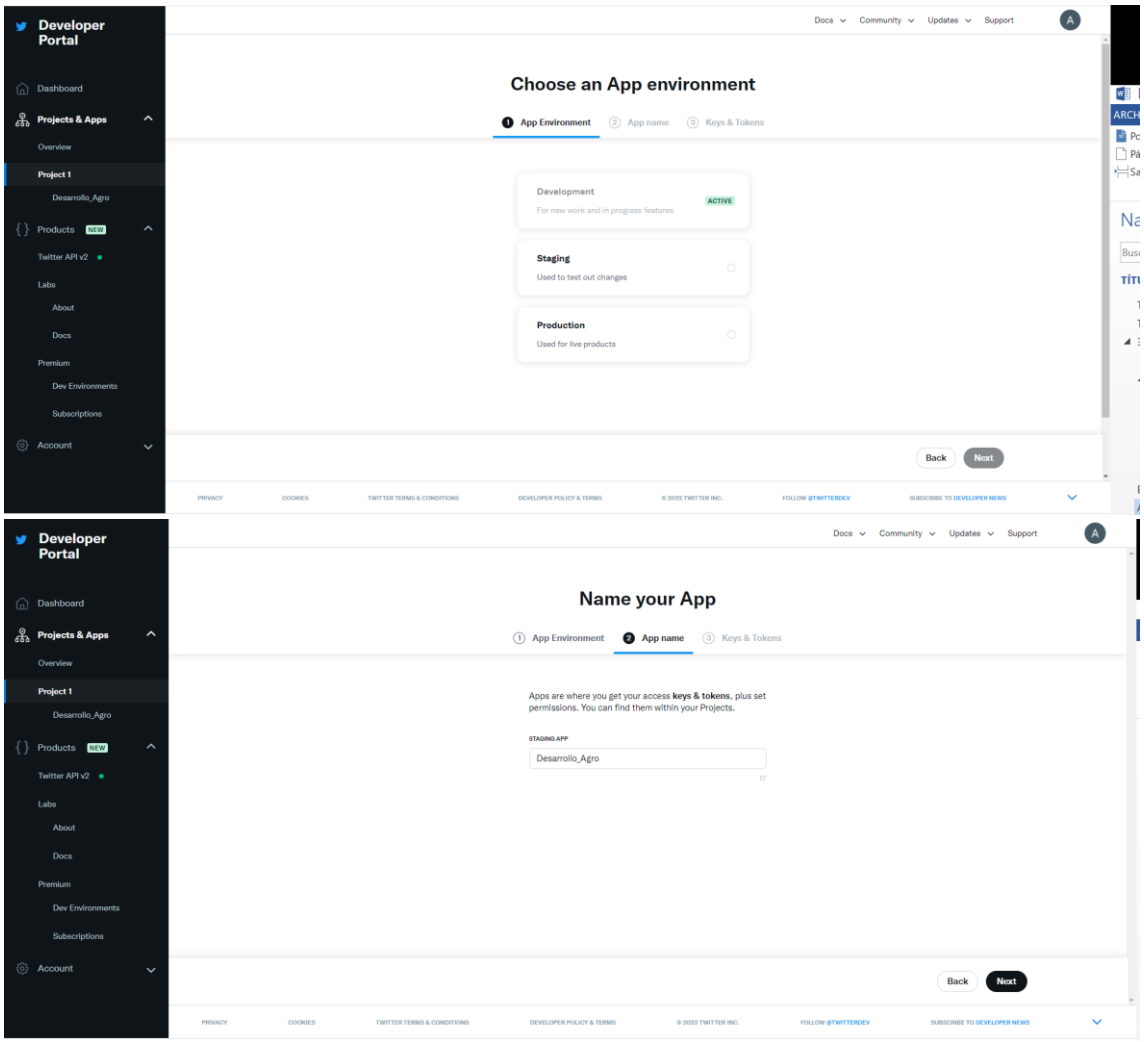


***Imagen 3**. Detalle del registro al Twitter Developer Portal (II).*
Fuente: Twitter Developer Portal.

- En la última ventana, aparecerán las claves y *tokens* que harán falta introducir durante el proceso de autenticación de las credenciales del usuario para extraer datos mediante la API. Estas claves son confidenciales y sólo aparecen una vez, el usuario debe guardarlas en un lugar seguro y no compartirlas.

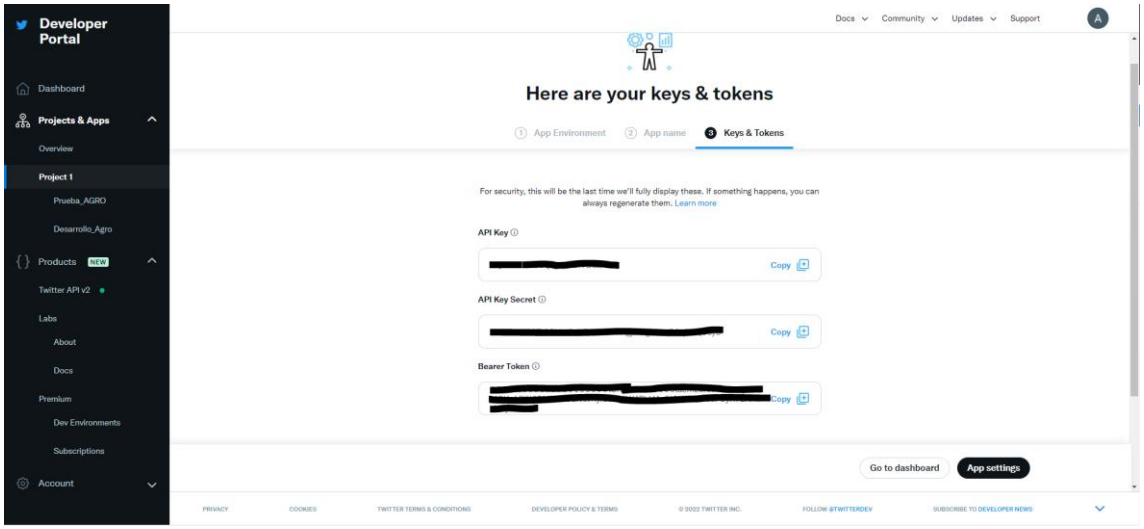


***Imagen 4**. Credenciales de acceso a la API de Twitter.*

Fuente: Twitter Developer Portal.

3. Acceso "Academic Research": para poder conseguir acceso elevado, el usuario debe dirigirse al área de "Productos" dentro del Portal del Desarrollador y después a "Twitter API" para rellenar la solicitud.

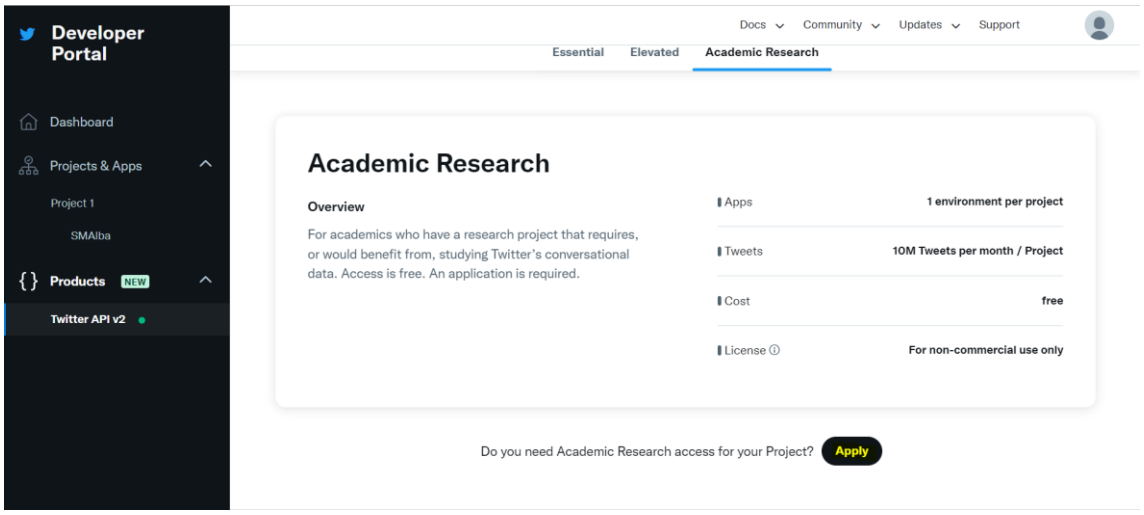


***Imagen 5**. Vista del Twitter Developer Portal.*
Fuente: Twitter Developer Portal.

- Una vez dentro de la solicitud, el usuario debe rellenar varias pestañas con datos personales e información que demuestre su perfil académico.

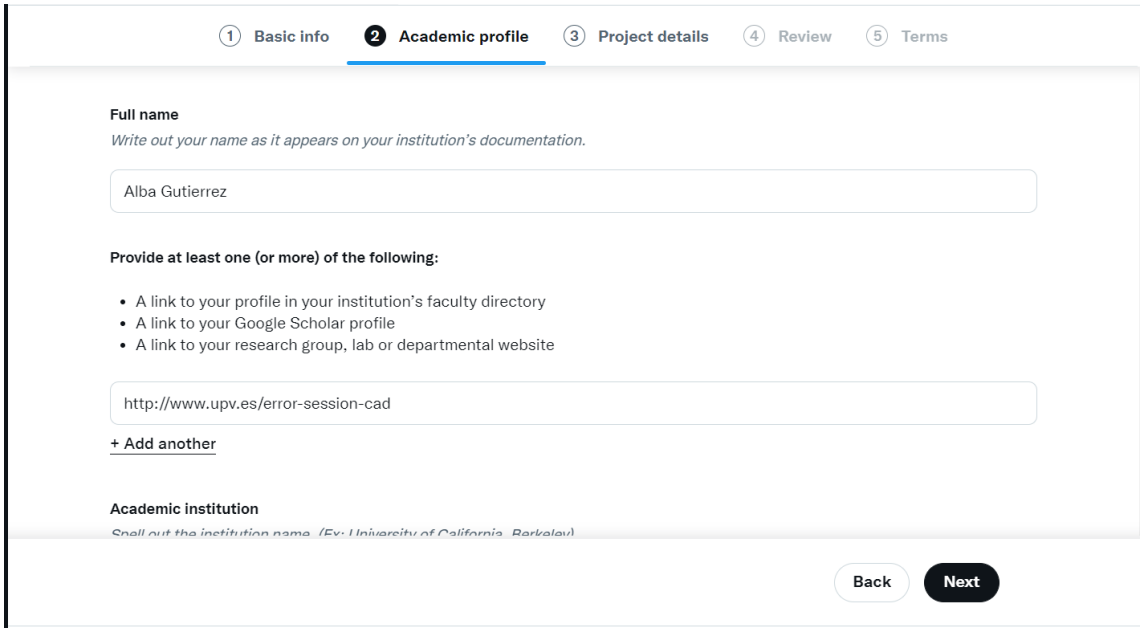


***Imagen 6**. Detalles de algunas de las respuestas al formulario de solicitud al acceso Academic Research.*

Fuente: Twitter Developer Portal.

- Acto seguido, deberá completar los detalles del proyecto para el que necesita extraer datos de la plataforma.

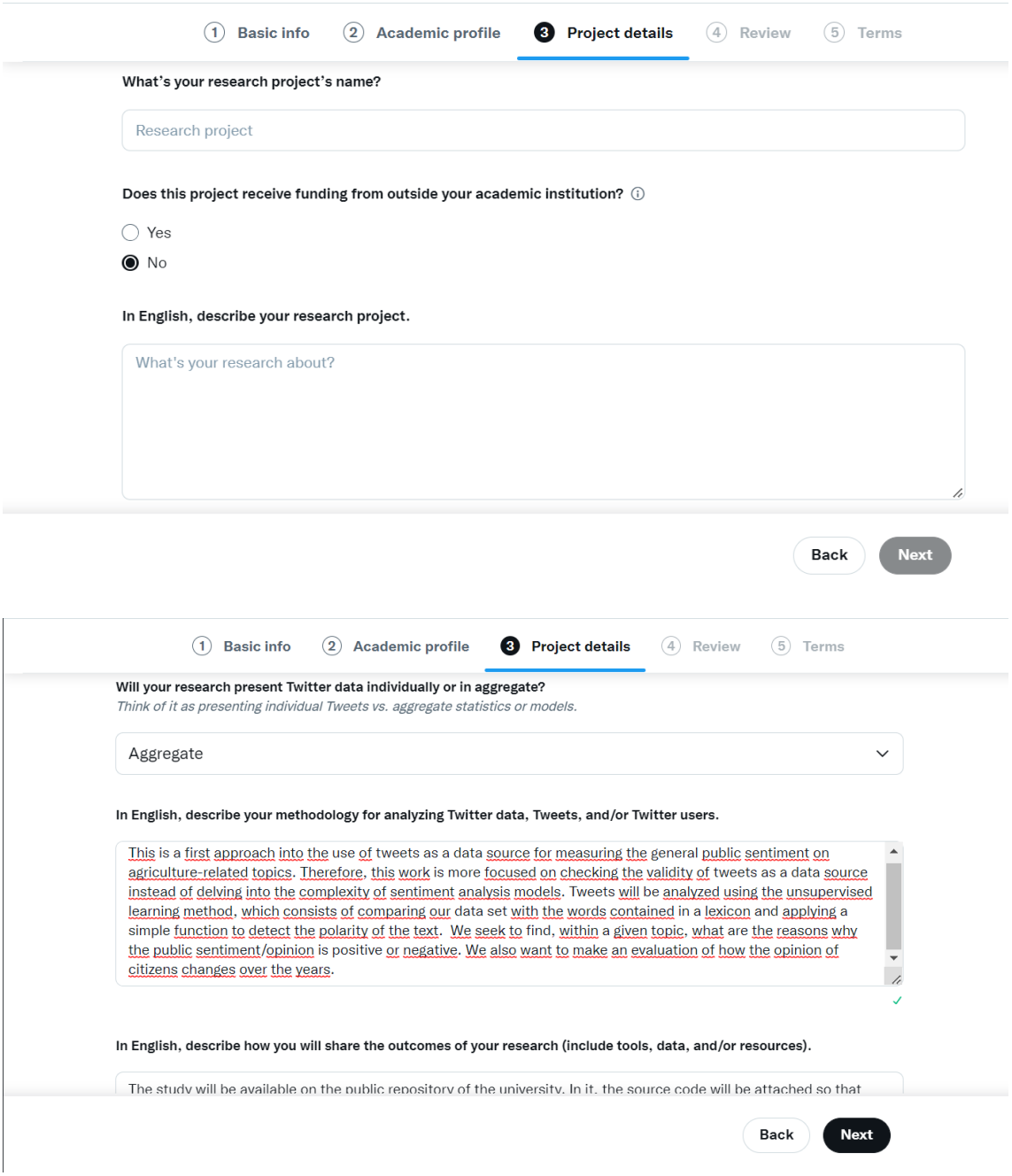
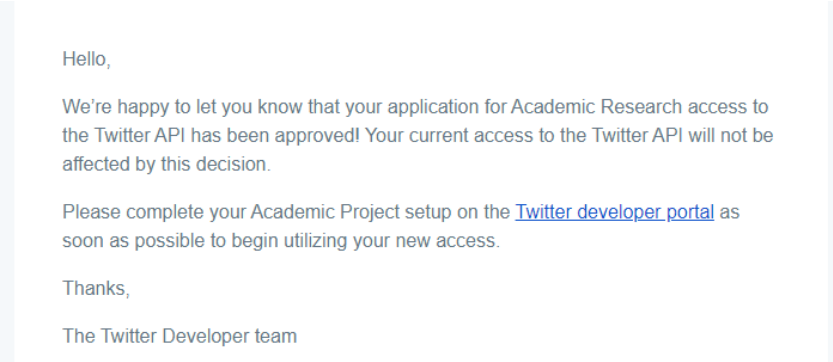


***Imagen 7**. Detalles de algunas de las respuestas al formulario de solicitud al acceso Academic Research (II). Fuente: Twitter Developer Portal.*

- Por último, firmará los términos y condiciones. Cada solicitud revisada en detalle por el equipo del Developer Portal. En caso de obtener una respuesta positiva, el perfil del usuario en el Developer Portar y su tipología de acceso cambiarán automáticamente.



Hello,

We're happy to let you know that your application for Academic Research access to the Twitter API has been approved! Your current access to the Twitter API will not be affected by this decision.

Please complete your Academic Project setup on the Twitter developer portal as soon as possible to begin utilizing your new access.

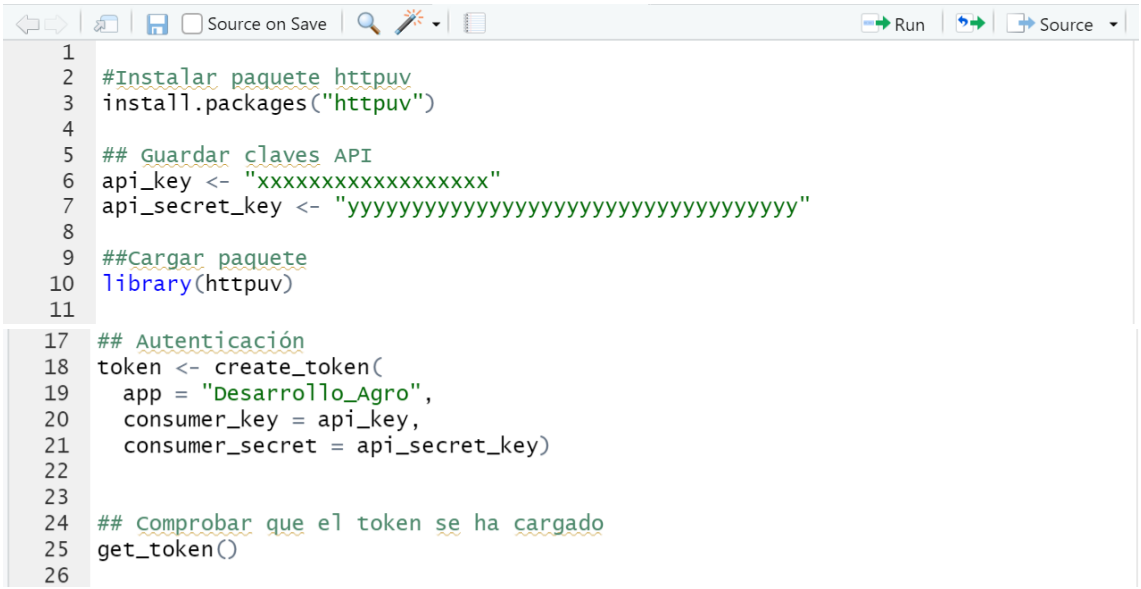
Thanks,

The Twitter Developer team

***Imagen 8**. Respuesta positiva del equipo de desarrolladores de Twitter.*
Fuente: Twitter Developer Portal.

4. Autorización via navegador web. Una vez que se ha creado la App de Twitter y el usuario conoce sus claves, es hora de conseguir autorización para acceder a la Twitter API. Para ello se ha de crear un *token* de acceso. Los *tokens* son *strings* aleatorios (cadenas de caracteres) que identifican a un usuario y pueden ser utilizados para realizar llamadas API. Los *tokens* expiran eventualmente y, en este caso, Twitter es capaz de ver qué aplicación generó el *token* (RStudio).

- En R, instalar y cargar los paquetes **httpuv**
- Copiar y pegar en R las claves (*API key* y *API secret key*) obtenidas al crear la App de Twitter.
- Autentificar con la función **create_token** del paquete **httpuv**



Source on Save | Run | Source

```
1
2  #Instalar paquete httpuv
3  install.packages("httpuv")
4
5  ## Guardar claves API
6  api_key <- "xxxxxxxxxxxxxxxx"
7  api_secret_key <- "yyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy"
8
9  ##Cargar paquete
10 library(httpuv)
11
17 ## Autenticación
18 token <- create_token(
19   app = "Desarrollo_Agro",
20   consumer_key = api_key,
21   consumer_secret = api_secret_key)
22
23
24 ## Comprobar que el token se ha cargado
25 get_token()
26
```

***Imagen 9**. Autenticación de las claves de acceso a la Twitter API.*
Fuente: RStudio.

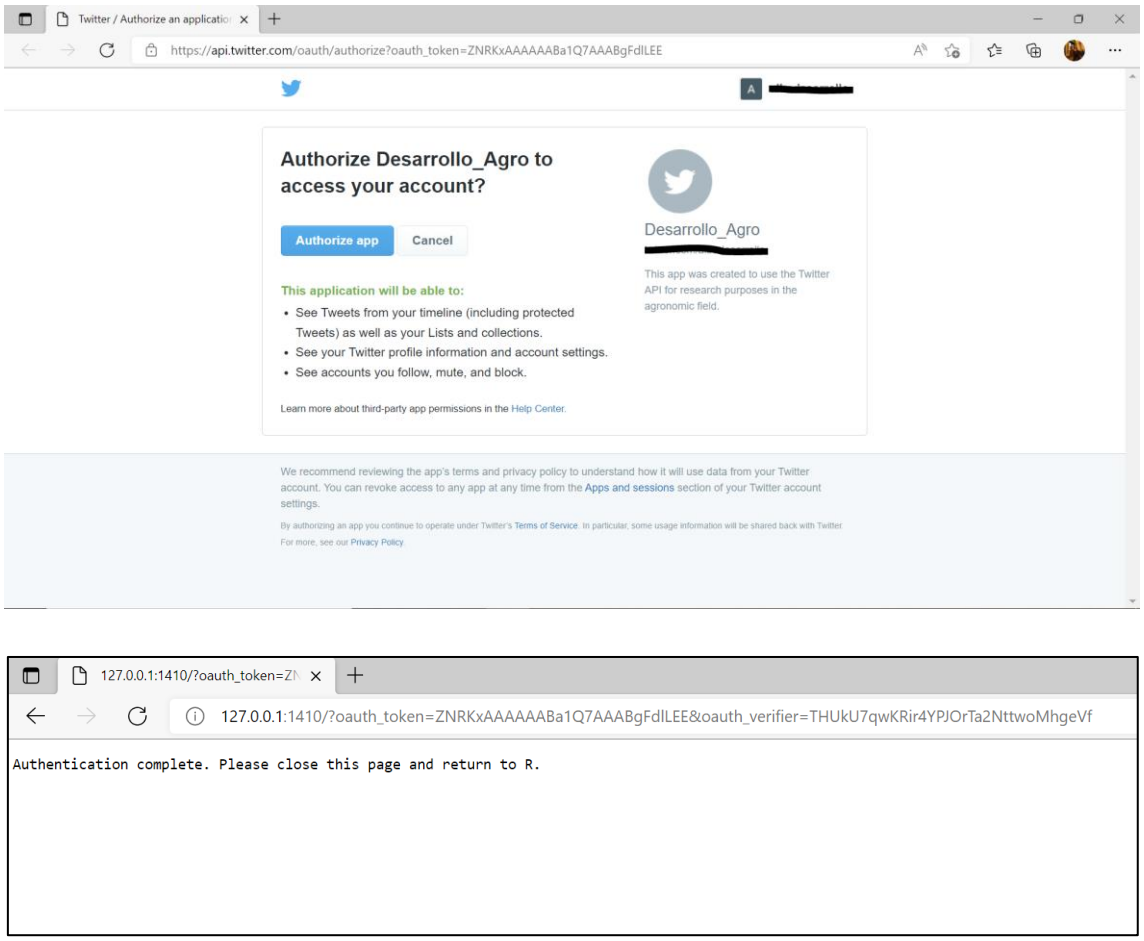


***Imagen 10**. Ventana de confirmación del acceso autorizado a la API de Twitter.*
Fuente: Twitter Developer Portal.

Una vez que aparece el mensaje de la Imagen 10, el acceso a la API de Twitter queda habilitado y pueden comenzar la extracción de *tuits* desde RStudio.

IV. Desarrollo del caso 1 (caso piloto)

Conjunto de datos:

La selección de la temática a analizar ha estado condicionada a varios limitantes. Principalmente, el volumen de *tuits* disponibles, el idioma y la geolocalización. La siguiente tabla recoge varias temáticas que fueron finalmente descartadas:

TEMÁTICA	IDIOMA	GEOLOCALI-ZACIÓN	KEY WORDS	Nº *TUITS*	FECHA
Organismos transgénicos	Español	España	"#transgénicos" "#OGM" "#GMOs" "genéticamente modificado"	**20**	01/05/2022
Agricultura ecológica	Español	España	"agricultura ecológica"	**33**	01/05/2022
Agricultura ecológica	Inglés	Holanda	"organic farming" "agroecology"	**23**	01/05/2022
Macrogranjas	Español	España	"macrogranjas"	**86**	01/05/2022

Tabla 2. *Posibles temáticas del caso 1 descartadas por disponer de una muestra demasiado pequeña.*
Fuente: Elaboración propia.

Todas las temáticas descartadas devolvieron un conjunto de *tuits* demasiado pequeño como para poder llevar a cabo una evaluación fiable (ver Tabla 2), debido al riesgo de obtener una alta imprecisión en los resultados.

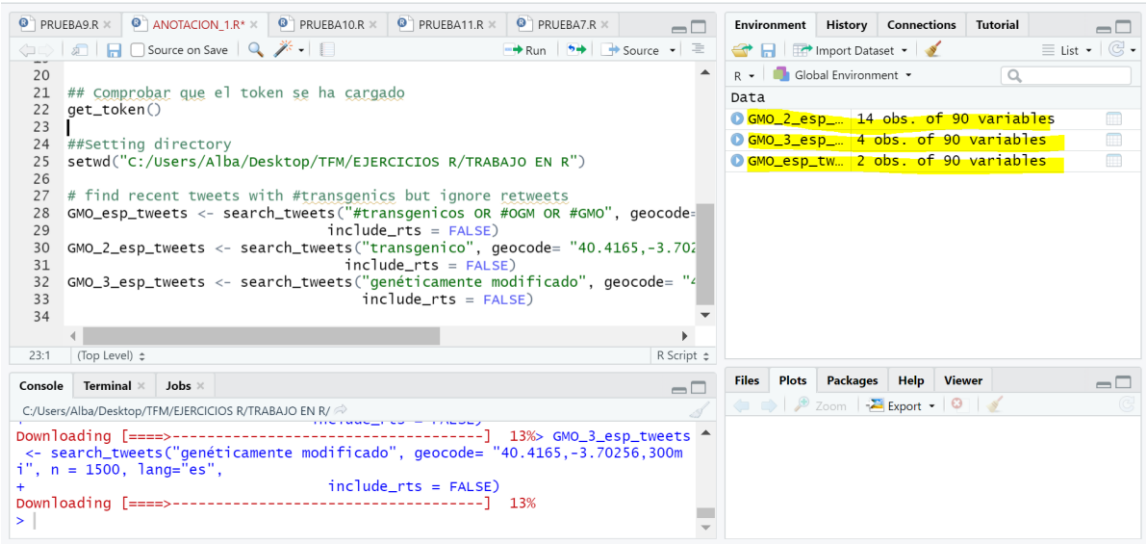


Imagen 11. *Muestra del escaso número de* tuits *publicados en España sobre la temática de transgénicos entre las fechas 23/04/2022 y 01/04/2022.*
Fuente: Captura de pantalla de RStudio.

Adicionalmente, la temática de las macrogranjas decidió descartarse al tratarse de una palabra con connotaciones inherentemente negativas, y por tanto los usuarios de Twitter que la emplean son en su mayoría detractores del sistema de ganadería intensiva (ver Tabla 3). Por tanto, no contamos con una muestra representativa del conjunto de la población al haber escogido una palabra clave poco adecuada.

DATE	USER	TEXT	TIDY TEXT	SENTIMENT
29/04/2022	TribunaVa	PACMA se manifiesta en contra de las macrogranjas de Corcos del Valle y Quintanilla de Trigueros	PACMA se manifiesta en contra de las macrogranjas de Corcos del Valle y Quintanilla de Trigueros	NEGATIVO
29/04/2022	CyLPACMA	La industria ganadera es altamente contaminante y contribuye a la despoblación. Por encima de todo está basada en la crueldad extrema y en el sufrimiento de animales.Hemos presentado alegaciones contra dos macrogranjas en Valladolid	La industria ganadera es altamente contaminante y contribuye a la despoblación. Por encima de todo está basada en la crueldad extrema y en el sufrimiento de animales.Hemos presentado alegaciones contra dos macrogranjas en Valladolid	NEGATIVO
29/04/2022	noticiascyl	PACMA presenta alegaciones contra macrogranjas que "ponen en peligro" a dos municipios de Valladolid	PACMA presenta alegaciones contra macrogranjas que "ponen en peligro" a dos municipios de Valladolid	NEGATIVO
29/04/2022	sylvitere	¡Macrogranjas NO!:ni en Caparroso ni en Noviercas ni en ninguna parte	¡Macrogranjas NO!:ni en Caparroso ni en Noviercas ni en ninguna parte	NEGATIVO
29/04/2022	TopoSudaka	#Macrogranjas en #España deforestación en #Brasil	#Macrogranjas en #España deforestación en #Brasil	NEGATIVO

Tabla 3. *Ejemplo de varios de los* tuits *sobre macrogranjas publicados en España, extraídos de Twitter el día 01/05/2022.*
Fuente: Captura de pantalla. RStudio.

Por todo ello, y reconociendo la ventaja de conocer la geolocalización del conjunto de *tuits*, se amplió la muestra a nivel europeo y se eligió la agricultura ecológica como tema de estudio en lengua inglesa, para así contar con un mayor número de *tuits* y obtener datos de distintas regiones europeas y no predominantemente de España.

TEMÁTICA	IDIOMA	GEOLOCALIZACIÓN	KEY WORDS	Nº *TUITS*	FECHA
Agricultura ecológica	Inglés	Europa	"organic farming" "agroecology"	**216**	14/05/2022
Agricultura ecológica	Inglés	Europa	"organic farming" "agroecology"	**237**	23/05/2022

Tabla 4. *Características de la muestra de* tuits *recogidos los días 14 y 23 de mayo de 2022..*
Fuente: Twitter.

Las palabras clave escogidas fueron "organic farming" y "agroecology", utilizadas indistintamente en inglés para hacer referencia a la agricultura ecológica. Para contar con una muestra de datos mayor, se han recogido datos dos veces de manera asíncrona, con una separación de 9 días entre ambas extracciones para no obtener *tuits* repetidos. Por ello, el periodo temporal de la muestra se extiende desde el día 06/05/2022 hasta el día 23/05/2022.

Pre-procesamiento de datos:

En este caso, el proceso de tratamiento de datos ha consistido en la anotación manual. Para poder proceder a la anotación, se han descargado los *dataframe* en formato *.csv* para poder abrirlos en Excel y de esta forma hacer más cómoda para el revisor la visualización de datos. El tamaño de la muestra final anotada será de 275 *tuits*. A menudo, en procesos de anotación manual se necesitan entre 100 y 300 unidades de texto para obtener un conjunto de datos confiable (Van Atteveldt, W. et al, 2021).

El proceso ha consistido en clasificar los *tuits* bajo 4 categorías distintas:

-**POSITIVE**: si el *tuit* reflejaba un sentimiento positivo hacia las prácticas agroecológicas, la agricultura ecológica, su legislación y prácticas asociadas como la permacultura, la agricultura regenerativa, el manejo integrado de plagas, etc.

-**NEGATIVE**: si el *tuit* denota un sentimiento claramente negativo hacia el conjunto de prácticas mencionadas en el párrafo anterior.

-**NEUTRAL/INDIFERENTE**: si el *tuit* habla acerca del "organic farming" y el "agroecology" pero no refleja ninguna opinión.

-**NOT RELATED**: si el *tuit* menciona las palabras clave pero en conjunto no trata sobre dicha temática, o bien refleja una opinión sobre otro tema distinto a la agricultura ecológica.

Posteriormente, para el análisis de la metodología *lexicon-based*, se descartan aquellos *tuits* clasificados dentro de la categoría de NOT RELATED, así como aquellos que están repetidos y otros que pueden conducir a error al modelo. En esta última categoría entrarían aquellos *tuits* que:

-Hacen referencia a la agricultura intensiva o convencional de forma positiva/negativa y por tanto confunden al modelo ya que este no puede distinguir si se está hablando de la agricultura ecológica o sobre otro tema.

-Emplean el sarcasmo.

-Emplean las comillas para indicar ironía.

-Muestran dos opiniones diferentes en el mismo *tuit*.

-Adjuntan una imagen, link, vídeo, etc. que condiciona el sentido del *tuit* y que el modelo no es capaz de procesar.

Algunos ejemplos de *tuits* extraídos que han sido descartados por la dificultad para interpretarlos se recogen en la siguiente tabla:

DATE	USER	TEXT	SENTIMENT	COMENTARIOS
10/05/2022 10:35	khknickel	Clash in Czechia over organic farming. Agribusinesses oppose EU plan to extend organic farming to 25% of the land. Small farmers eye EU plans as opportunity. Rising prices for pesticides and fertilisers mean farmers must look for alternatives anyway	-	Se exponen dos opiniones diferentes.
10/05/2022 18:30	ThomasPoetter	It is worth considering the experience of Sri Lanka when we thinking about "organic" agriculture	NEGATIVE	Las comillas invierten el sentido de la frase.
12/05/2022 10:15	natmakar	Yes, by and large industrial farming is currently feeding the world. This is not my point. My point is: this is not sustainable, we know it, and unjustly bashing organic farming isn't a way to solve this problem.	POSITIVE	Hace referencia a la agricultura intensiva de forma negativa, no la orgánica.
16/05/2022	transitionlouth	I've been pointing out for half a century that organic farming is the way to go. Unfortunately, having it suddenly imposed on a chemicals based farming system with no time for transition will mean some of the poorest people will starve."	NEUTRAL	Se exponen dos opiniones diferentes.

19/05/2022	muck_real	Yes, let's double the grants for farmers in England to convert to land-hungry, inefficient organic farming #braindead	NEGATIVE	Ironía.
19/05/2022	gremlin100	So, organic farming is worse than anything? What a ridiculous, unproven, disrespectful comment to make, you make these sensationalist comments because, oh, you have a book coming out! You keep writing your drivel, we'll keep producing top quality food whilst enhancing our ground	POSITIVE	Ironía. Hace comentarios negativos no dirigidos hacia la agricultura ecológica.

Tabla 5. *Muestra de* tuits *sobre agricultura ecológica que pueden conducir a error al modelo* lexicon-based*.*
Fuente: Twitter.

Tras descartar los grupos mencionados, el conjunto de *tuits* se reduce a un total de 275, de los cuales:

NEGATIVO	NEUTRO	POSITIVO
64	*103*	*108*
23%	**37%**	**39%**

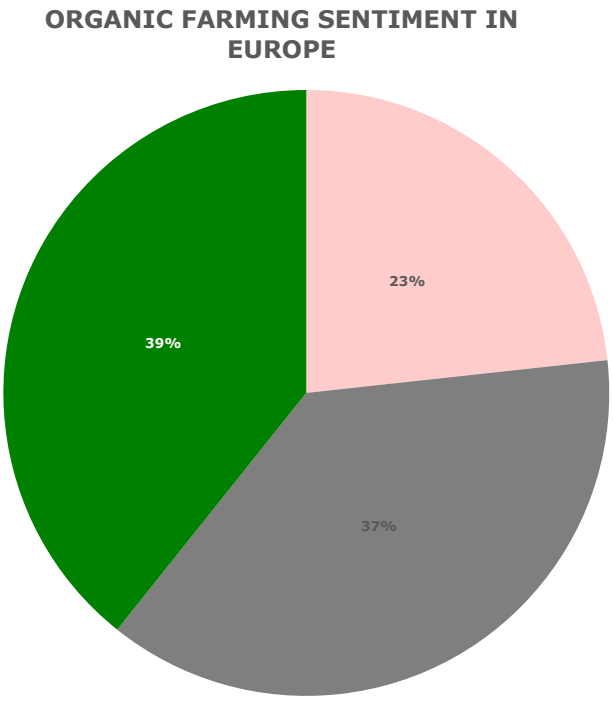


Imagen 12. *Gráfico de distribución del sentimiento hacia la agricultura ecológica en Europa. Método de anotación manual.*
Fuente: Elaboración propia en Excel.

Testeo:

Para la metodología de aprendizaje no supervisado, se emplearon como base de referencia tres lexicon distintos y posteriormente se determinó con cuál de ellos se obtiene un mejor resultado, en comparación con el proceso de anotación manual. Los *lexicon* son los siguientes:

1. El *opinion lexicon* de los autores Hu and Liu (2004). Se trata de una lista de más de 6.800 palabras de opinión (positivas y negativas) en inglés, recopiladas por los autores a lo largo de muchos años.

2. El *lexicon* Afinn, del autor Finn Nielsen. Se encuentra disponible en el repositorio Github, por lo que es fácilmente accesible. En este caso, además de una ristra de más de 2.400 palabras de connotación positiva o negativa, cada una de ellas está clasificada dentro de un rango con un número entero entre menos cinco (negativo) y más cinco (positivo). Hay disponibles múltiples versiones, para el presente trabajo se escogió la versión Afinn-111.

3. El lexicon Jocker, del autor Matthew L. Jockers. El paquete ***sentimentr*** se va a emplear como tercer método de análisis de opinión. Este paquete tiene asociado por defecto el *lexicon* Jocker, también llamado *syuzhet* (Rinker, 2021). Este lexicon fue desarrollado por el Nebraska Literary Lab y contiene más de 10.748 palabras. Al igual que el lexicon Afinn, todas las palabras tienen asignado un valor de sentimiento, que abarca un rango de entre −1 y 1.

Salvo en el caso de ***sentimentr*** en el que se han utilizado las funciones predeterminadas ***sentiment*** y ***sentiment_by*** para obtener el sentimiento agregado de cada *tuit*, para el análisis con los *lexicon* Afinn y Hu and Liu se ha creado una función en base a distintos casos de uso[1][2][3] sobre el análisis de opinión en Twitter, ajustándola a las condiciones particulares de este estudio. El objetivo es crear una función que calcule la 'puntuación' de cada *tuit* en función del número de veces que aparezcan palabras de sentimiento positivo o negativo en el mismo. Cada vez que una palabra de este tipo aparezca en el texto (siendo este el conjunto de *tuits*) se sumará un punto a la puntuación final o bien se restará, en función de si dicha palabra es de tipo positivo (+1) o bien de tipo negativo (-1). La función tomará como base de datos las palabras recogidas en el *lexicon* que se haya cargado previamente en RStudio. La función se puede consultar en el código R del caso 1, aportado en el Anexo V.

Resultados:

El empleo de la función con el *lexicon* Hu and Liu devolvió los resultados más próximos a los obtenidos mediante anotación manual. Del conjunto de *tuits* analizados, cataloga a un 48% como positivos, un 28,7% como neutrales y un 23,3% como negativos. Se observa una desviación de 9 puntos respecto a la anotación manual entre las categorías "Positive" y "Neutral", mientras que no se observan desviaciones en el porcentaje de *tuits* negativos respecto a los resultados obtenidos manualmente.

[1] https://towardsdatascience.com/twitter-sentiment-analysis-and-visualization-using-r-22e1f70f6967
[2] https://smutuvi.github.io/blog/2017/03/03/twitter-data-analysis-with-r/
[3] https://medium.com/@joexu_34923/perceptions-toward-the-dallas-fuel-d180071ad328

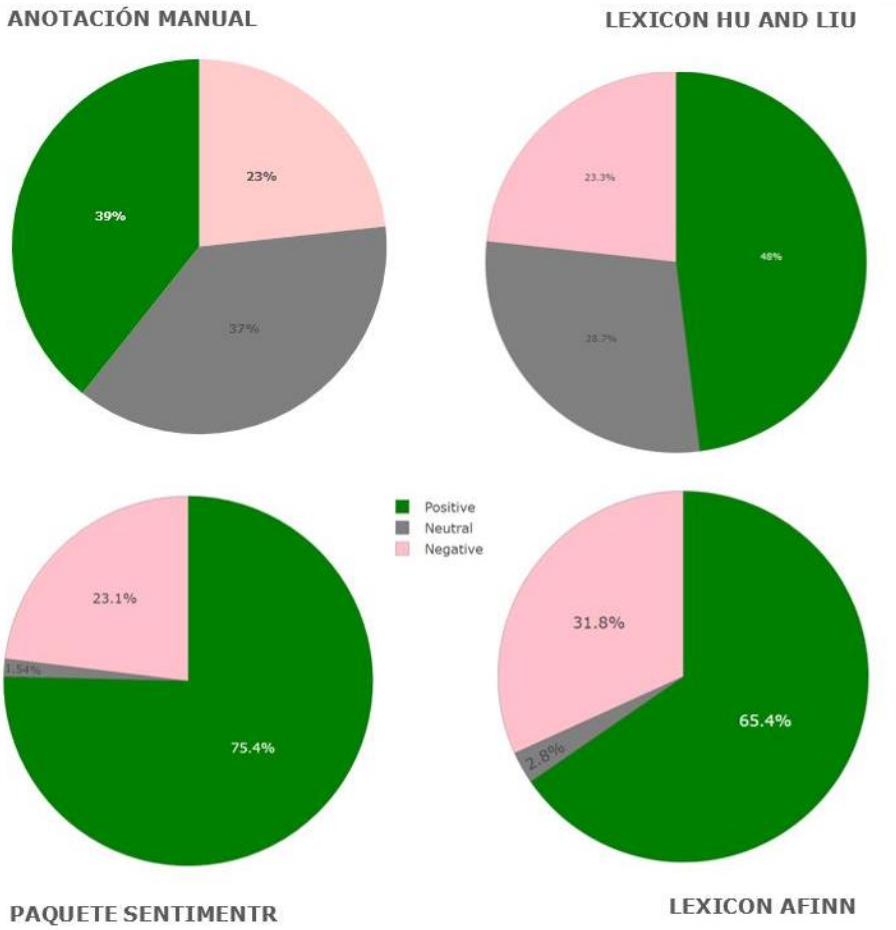


Imagen 13. *Resultados en R utilizando la metodología de aprendizaje no supervisado. La gráfica superior izquierda muestra los resultados de la anotación manual, mientras que la de la derecha muestra los resultados empleando el lexicon Hu and Liu. La gráfica inferior izquierda muestra los resultados con el paquete sentimentr y la gráfica inferior derecha los resultados con el lexicon Afinn.*
Fuente: Elaboración propia en RStudio.

Por el contrario, las otras dos metodologías devolvieron unos resultados muy alejados de la clasificación manual. En concreto, se observa que en ambos casos el modelo encuentra dificultades para identificar la neutralidad y tiende principalmente a categorizar esos *tuits* como positivos. **Por tanto, se descarta el uso de ambos para este estudio y se apuesta por el empleo del *lexicon* Hu and Liu para continuar con el análisis.**

Habiendo clasificado la muestra en distintas categorías de sentimiento, es posible extraer conclusiones acerca de las opiniones que muestran los usuarios de Twitter y lo que las motivan. A continuación, se analizaron cuáles fueron las palabras más mencionadas dentro de los grupos de *tuits* categorizados como "positive" y "negative". De esta manera, es posible conocer cuáles son los términos que la ciudadanía asocia más habitualmente con la agricultura ecológica, qué ventajas encuentran en este modelo de producción y asimismo cuáles son los problemas que le atribuyen (véanse imágenes 14 y 15, gráficos 2 y 3).

Con aquellos *tuits* que reflejaron una opinión positiva acerca de la agricultura ecológica, se creó una nube de palabras mediante la función ***rquery.wordcloud*** (del paquete ***wordcloud***). Las nubes de palabras son recursos visuales, muy habituales como

herramientas de *marketing*, que se utilizan para representar las palabras más destacadas que componen un texto. De forma abstracta, estas nubes presentan en un mayor tamaño aquellas palabras que aparecen con más frecuencia dentro del texto, permitiendo visualizar de manera rápida aquellos términos más representativos del contenido del texto. Para evitar que aparezcan dentro de la nube palabras irrelevantes, la función ***rquery.wordcloud*** permite eliminar del texto aquellos términos más comunes en la lengua inglesa (artículos, conjunciones, onomatopeyas, etc.). Adicionalmente, se configuró la función de manera que sólo aparecieran las palabras mencionadas un mínimo de 4 veces en el conjunto del texto y asimismo se eliminaron del análisis las palabras clave empleadas durante la recogida de datos ("organic farming" y "agroecology"), puesto que todos los *tuits* mencionaban al menos una de ellas, distorsionando el gráfico final.

Al analizar mediante la función ***rquery.wordcloud*** el conjunto de *tuits* de agricultura ecológica que mostraban un sentimiento positivo, el programa devolvió el siguiente resultado (para comprobar el análisis paso a paso, consultar el código en R del caso 1 en el Anexo V):

Imagen 14. *Nube de palabras del conjunto de* tuits *de sentimiento positivo sobre agricultura ecológica. Fuente: RStudio.*

Los resultados demuestran que la ciudadanía identifica a la agricultura ecológica como un modelo de producción de fuerte carácter "verde". Como se puede observar, los usuarios con una opinión positiva de la agricultura ecológica relacionaron esta práctica con términos como "sustainable" y "sustainability", "biodiversity" y "bees", "soil" o "health". Esta percepción coincide con recientes investigaciones científicas en este ámbito que refuerzan la visión de la producción ecológica como un modelo sostenible a largo plazo ("sustainable" y "sustainability"), como el estudio de Gabriel et al. (2010) sobre el impacto de la agricultura ecológica sobre la biodiversidad floral y de fauna ("bees", "biodiversity", "soil") en ecosistemas agrarios a distintas escalas, o este otro de Gattinger et al. (2012) que confirma que las prácticas orgánicas aumentan la concentración de carbono orgánico (SOC, por sus siglas en inglés) en las capas más superficiales del suelo respecto a los sistemas agrarios convencionales. Asimismo, la agricultura ecológica se relaciona con un menor número de

residuos químicos encontrados en productos como cereales (Rekha, Naik & Prasad, 2006) y frutas (Turgut, Ornek & Cutright, 2011), así como con una mejora de las propiedades nutricionales y organolépticas de los productos (Huber et al., 2011).

Con el paquete ***wordcloud*** es posible también obtener estos resultados como un gráfico de barras. En este caso, el gráfico 2 muestra únicamente las 10 palabras más mencionadas en el texto.

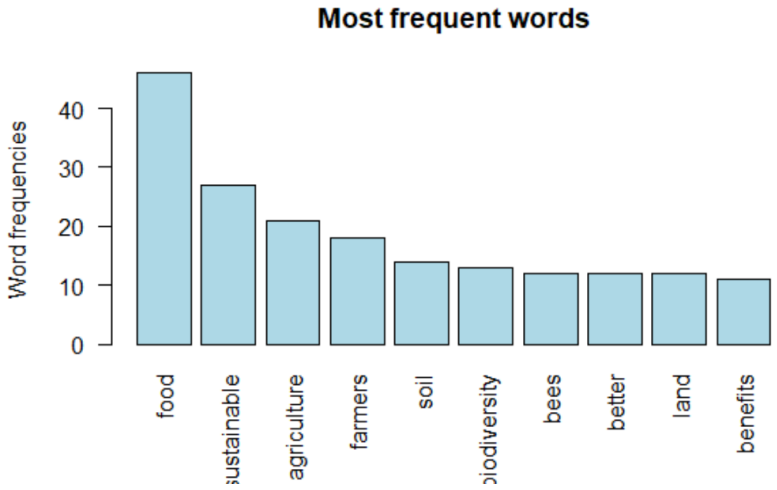


Gráfico 2. *Gráfico de barras con las 10 palabras más mencionadas en el conjunto de* tuits *sobre agricultura ecológica de categoría "Positive"..*
Fuente: RStudio.

En cuanto al conjunto de *tuits* de sentimiento negativo, las palabras recogidas en la nube fueron, en general, muy distintas (para comprobar el análisis paso a paso, consultar el código en R del caso 1 en el Anexo V):

Imagen 15. *Nube de palabras del conjunto de* tuits *de sentimiento negativo sobre agricultura ecológica.*
Fuente: RStudio.

En este caso, los usuarios críticos con la agricultura ecológica mencionan otros términos como "crisis", "prices", "yields", "food security" o "loss". Llama especialmente la atención el elevado número de menciones a Sri Lanka, una nación que se ha visto sacudida por una grave crisis alimentaria, en parte provocada por el veto del gobierno al uso de fertilizantes sintéticos en agricultura, reduciendo los rendimientos de los cultivos en hasta un 30% en todo el país (Reuters, 2022). **Actualización**: el 09 de julio de 2022, la población de la isla del Índico asaltó el palacio presidencial provocando la dimisión del presidente y del primer ministro srilakenses.

De nuevo, puede ser muy interesante aprovechar los resultados que devuelve el modelo para analizar qué preocupa a la ciudadanía en relación con la agricultura ecológica, qué aspectos percibe como puntos débiles de este modelo productivo. A simple vista, ya se puede apreciar cómo generalmente, los usuarios asocian la agricultura ecológica con rendimientos más bajos y con pérdidas ("yields", "loss"). También, con precios más altos y con una mayor inseguridad alimentaria ("prices", "food security"), algo que está directamente relacionado con los menores rendimientos que se obtienen con estas prácticas por norma general. Para un asesor político, esta información puede resultar útil a la hora de identificar qué posibles riesgos existen si se desea fomentar la incorporación de prácticas agroecológicas entre los productores y de esta forma, adelantarse a futuras complicaciones diseñando una estrategia coherente de mitigación y prevención de impactos.

En este caso, el gráfico de barras muestra de nuevo las 10 palabras más mencionadas en el conjunto de *tuit* (se debe tener en cuenta que Sri Lanka aparece dos veces por tratarse de un nombre compuesto).

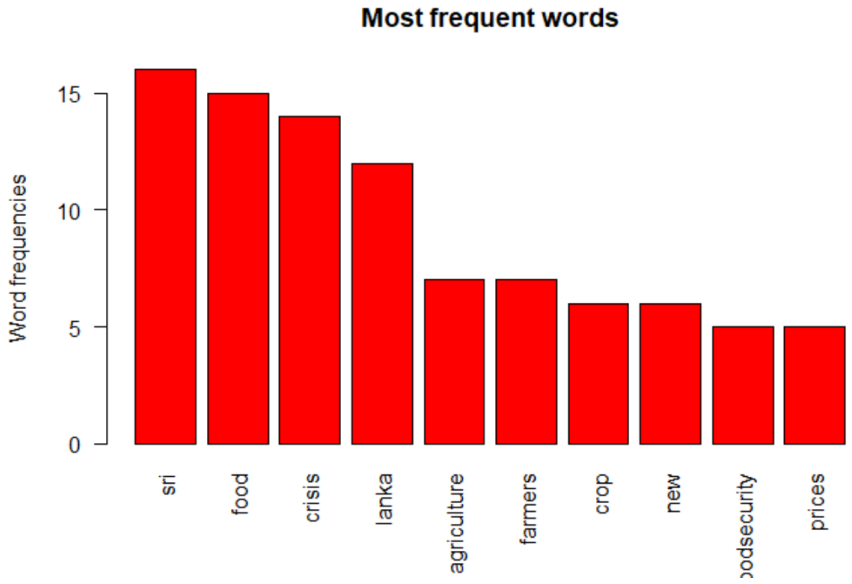


Gráfico 3. *Gráfico de barras con las 10 palabras más mencionadas en el conjunto de* tuits *sobre agricultura ecológica de categoría "Negative".*
Fuente: RStudio.

V. Código R del Caso 1

```
##AGRICULTURA ECOLÓGICA A NIVEL EUROPA EN INGLÉS

## Guardar claves API
api_key <- "[redacted]"
api_secret_key <- "[redacted]"

##Cargar paquetes necesarios
library(httpuv)
library(rtweet)
library(ROAuth)
library(plyr)
library(dplyr)
library(stringr)

## Autenticación
token <- create_token(
  app = "Desarrollo_Agro",
  consumer_key = api_key,
  consumer_secret = api_secret_key)


## Comprobar que el token se ha cargado
get_token()

##Establecer ruta
setwd("C:/Users/[redacted]")


#PRIMERA EXTRACCIÓN DE TWEETS SOBRE AGRICULTURA ECOLOGICA EN INGLES Y EN TODA
EUROPA
#FECHA 14/05/2022
orgeurope <- search_tweets("organic farming", geocode=
"50.117286,9.247769,900mi", n = 1500, lang="en",
                           include_rts = FALSE)
orgeurope2 <- search_tweets("agroecology", geocode=
"50.117286,9.247769,900mi", n = 1500, lang="en",
                            include_rts = FALSE)

#Quitar columnas que no nos interesan
#Dejar solo fecha, texto y nombre de usuario
orgeurope_tidy <- orgeurope %>% select(created_at,screen_name,text,)
orgeurope2_tidy <- orgeurope2 %>% select(created_at,screen_name,text,)


#Juntar ambos dataframes en uno solo
tidy_as <-as.data.frame(orgeurope_tidy)
tidy2_as <-as.data.frame(orgeurope2_tidy)
orgeur_merge <- merge(x=tidy_as, y=tidy2_as, all = TRUE)

#Pasarlo a formato csv y guardarlo en el ordenador para la anotación manual
write.csv2(orgeur_merge,"Anotacion_orgeur")


#SEGUNDA EXTRACCIÓN DE TWEETS SOBRE AGRICULTURA ECOLOGICA EN INGLES Y EN TODA
EUROPA
#FECHA 23/05/2022
```

```
Orgeurope3 <- search_tweets("organic farming", geocode=
"50.117286,9.247769,900mi", n = 1500, lang="en",
                              include_rts = FALSE)
orgeurope4 <- search_tweets("agroecology", geocode=
"50.117286,9.247769,900mi", n = 1500, lang="en",
                               include_rts = FALSE)

#Quitar columnas que no nos interesan
#Dejar solo fecha, texto y nombre de usuario
orgeurope3_tidy <- orgeurope3 %>% select(created_at,screen_name,text,)
orgeurope4_tidy <- orgeurope4 %>% select(created_at,screen_name,text,)


#Juntar ambos dataframes en uno solo
tidy3_as <-as.data.frame(orgeurope3_tidy)
tidy4_as <-as.data.frame(orgeurope4_tidy)
orgeur_merge <- merge(x=tidy3_as, y=tidy4_as, all = TRUE)

#Pasarlo a formato csv y guardarlo en el ordenador para la anotación manual
write.csv2(orgeur_merge,"Anotacion_orgeur2")


##Función para la detección del sentimiento
#Primero el pre-procesamiento o limpieza de los tuits
score.sentiment <- function(tweets, pos.words, neg.words, .progress='none')
{
  require(plyr)
  require(stringr)
  scores <- laply(tweets, function(tweet, pos.words, neg.words)
  {
    tweet = gsub('https://','', tweet)      # eliminar links https://
    tweet = gsub('http://','', tweet)       # eliminar links http://
    tweet = gsub('[[:punct:]]','', tweet)   # eliminar símbolos de puntuación
    tweet = gsub('[[:cntrl:]]','', tweet)   # eliminar caracteres de control
    tweet = gsub('\\d+','', tweet)          # eliminar números
    tweet = gsub('[^[:graph:]]',' ', tweet)  # reemplazar emojis con espacios
    tweet = str_replace_all(tweet,"[^[:graph:]]", " ")
    tweet = tolower(tweet)                  #Transformar texto a minúsculas
    word.list = str_split(tweet, '\\s+')           #Dividir cada tuit por
palabras
    words = unlist(word.list)                       #Transformar la lista de
palabras en un vector
    ##Emparejar las palabras positivas/negativas del dataset con el lexicon
    pos.match = match(words,pos.words)
    neg.match = match(words,neg.words)
    #Convertir valores coincidentes en TRUE o FALSE (1 and 0)
    pos.match = !is.na(pos.match)
    neg.match = !is.na(neg.match)
    #Obtener puntuaciones en base a la suma de valores coincidentes
    score <- sum(pos.match) - sum(neg.match)
    return(score)
  }, pos.words, neg.words, .progress=.progress)
  scores.df <- data.frame(score=scores, text=tweets)
  return(scores.df)
}

##Cargar el lexicon (Hu and Liu, KDD-2004)
pos.words <- scan("C:/Users/[redacted]/POSITIVE_LEXICON_eng.txt",
what="character")
```

```
neg.words <- scan("C:/Users/[redacted]/NEGATIVE_LEXICON_eng.txt",
what="character")

#Subimos los dataframe anotados manualmente
install.packages("readxl")
library(readxl)
ruta_excel <- "C:/Users/[redacted]/Annotated_orgtweets.xlsx"
ruta_excel2 <- "C:/Users/[redacted]/Annotated_org2.xlsx"
Annotated_tweets= read_excel(ruta_excel)
Annotated= read_excel(ruta_excel2)

#APLICAMOS LA FUNCIÓN EN LOS DOS DATAFRAMES ANOTADOS
TwtEcoEUR=score.sentiment(tweets = Annotated_tweets$TEXT,
pos.words,neg.words,.progress = 'text')
TwtOrg=score.sentiment(tweets = Annotated$TEXT, pos.words,neg.words,.progress
= 'text')


#Crear nueva variable en los dataframe que muestre la categoría de
sentimiento
TwtEcoEUR$score.cat <- ifelse(TwtEcoEUR$score > 0, "Positive",
                              ifelse(TwtEcoEUR$score < 0, "Negative",
"Neutral"))
TwtOrg$score.cat <- ifelse(TwtOrg$score > 0, "Positive",
                              ifelse(TwtOrg$score < 0, "Negative",
"Neutral"))

#Juntar ambos dataframes en uno solo
merge_1 = as.data.frame(TwtEcoEUR)
merge_2 = as.data.frame(TwtOrg)
combination = merge(x=merge_1, y=merge_2, all = TRUE)


#AHORA PODEMOS MEDIR EL SENTIMIENTO GLOBAL DE LA MUESTRA
##Aislar las distintas categorías y contabilizar los resultados
POSP <- length(which(combination$score.cat ==
"Positive"))/length(combination$score.cat)
NEUP <- length(which(combination$score.cat ==
"Neutral"))/length(combination$score.cat)
NEGP <- length(which(combination$score.cat ==
"Negative"))/length(combination$score.cat)

##Combinar valores calculados en un nuevo dataframe
W3trial <- data.frame(PRP = c(POSP, NEUP, NEGP), cat = c("Positive",
"Neutral", "Negative"))


#Crear un gráfico circular con la librería plotly
library('plotly')
W3trial$cat <- factor(W3trial$cat,
                      levels = c("Positive", "Neutral", "Negative"))
WIPx <- plot_ly(W3trial, labels = W3trial$cat, values = W3trial$PRP, type =
'pie',
                sort = FALSE,
                direction = "clockwise",
                textposition = 'inside',
                textinfo = 'label + percent',
                insidetextfont = list(color = '#FFFFF'),
                marker = list(colors = c('green', 'grey', 'pink'),
```

```
                              line = list(color = '#FFFFF', width = 0.25)))
%>%
  layout(title = 'Global organic farming European Sentiment', showlegend =
TRUE,
         xaxis = list(showgrid = FALSE, zeroline = FALSE, 'showticklabels' =
FALSE),
         yaxis = list(showgrid = FALSE, zeroline = FALSE, 'showticklabels' =
FALSE))
```

WIPx

```
##AHORA VEMOS LAS PALABRAS QUE MÁS SE MENCIONAN EN LOS TUITS

#Ordenar las filas según la categoría de sentimiento y extraer dos
dataframes, el de tuits positivos y el de tuits negativos
df_ordenado <- combination[order(combination$score.cat), ]

Negative_tweets <- df_ordenado[1:64,]
Positive_tweets <- df_ordenado[144:275,]

#Frecuencia de palabras
#Paquetes necesarios
library(tidyverse)
library(RColorBrewer)
library(wordcloud)
library(tm)
library(SnowballC)
library(RCurl)
library(XML)

Freq_Pos = str_replace_all(Positive_tweets$text, "@\\w+", "")
wordCorpus = Corpus(VectorSource(Freq_Pos))
wordCorpus <- tm_map(wordCorpus, removePunctuation)
wordCorpus <- tm_map(wordCorpus, content_transformer(tolower))
wordCorpus <- tm_map(wordCorpus, removeWords, stopwords("english"))
wordCorpus <- tm_map(wordCorpus, removeWords, c("amp", "2yo", "3yo", "4yo"))
wordCorpus <- tm_map(wordCorpus, stripWhitespace)

#Quitamos del analisis las palabras clave
#Quitamos del análisis los stop words que han pasado el filtro de limpieza
previo
wordCorpus <- tm_map(wordCorpus, removeWords,
c("organic","farming","agroecology","will","can"))

wcpos <- rquery.wordcloud(x=wordCorpus, type ="text",
                          lang = "english",
                          excludeWords = NULL,
                          textStemming = FALSE,
                          colorPalette = "Set1",
                          min.freq = 4,
                          max.words = 50)


#Gráfico de barras con la función wordcloud
#Se extraen las 10 palabras más mencionadas en los tuits
tdm <- wc2$tdm
freqTable <- wc2$freqTable
head(freqTable, 10)
```

```
barplot(freqTable[1:10,]$freq, las = 2,
        names.arg = freqTable[1:10,]$word,
        col ="lightblue", main ="Most frequent words",
        ylab = "Word frequencies")

#AHORA LO MISMO PARA LOS TUITS NEGATIVOS
#Frecuencia de palabras

Freq_Neg = str_replace_all(Negative_tweets$text, "@\\w+", "")
wordCorpus3 = Corpus(VectorSource(Freq_Neg))
wordCorpus3 <- tm_map(wordCorpus3, removePunctuation)
wordCorpus3 <- tm_map(wordCorpus3, content_transformer(tolower))
wordCorpus3 <- tm_map(wordCorpus3, removeWords, stopwords("english"))
wordCorpus3 <- tm_map(wordCorpus3, removeWords, c("amp", "2yo", "3yo",
"4yo"))
wordCorpus3 <- tm_map(wordCorpus3, stripWhitespace)

#Quitamos del analisis las palabras clave
#Quitamos del análisis los stop words que han pasado el filtro de limpieza
previo
wordCorpus <- tm_map(wordCorpus3, removeWords,
c("organic","farming","agroecology","will","can", "dont", "another"))

wcneg <- rquery.wordcloud(x=wordCorpus3, type ="text",
                     lang = "english",
                     excludeWords = NULL,
                     textStemming = FALSE,
                     colorPalette = "Set1",
                     min.freq = 4,
                     max.words = 50)


#Gráfico de barras con la función wordcloud
#Se extraen las 10 palabras más mencionadas en los tuits
tdmneg <- wcneg$tdm
freqTable2 <- wcneg$freqTable
head(freqTable2, 10)
barplot(freqTable2[1:10,]$freq, las = 2,
        names.arg = freqTable2[1:10,]$word,
        col ="red", main ="Most frequent words",
        ylab = "Word frequencies")
```

VI. *Dataframe* del Caso 1

El *dataframe* al completo no se ha incluido por razones de espacio. Cualquier interesado puede solicitarlo a la autora.

Fecha	Usuario	Texto	Categoría
06/05/2022	RichardSpeaks24	Reality is we should make a law which allows free range and organic and the rest be banned. No to battery farming, no to intensive farming, no to pesticides and spraying which enters water courses. Government must take action to steward land-use better and regrow hedgerows.	Positive
06/05/2022	mynatorigins	Faced with the increasing expectations of consumers for #transparency and #naturalness, #organic farming provides many opportunities for #nutraceutical and #agrifood industries. What about the consumer choice criteria? How to choose a supplier?	Positive
06/05/2022	TPOrganics	Build up regional grain reserves and global #foodsecurity response systems:Diversify food production and trade,Rebuild resilience and cut harmful dependencies through #agroecology as a form of crisis response, a route to resilience and a low-cost way to hedge against various shocks	Positive
06/05/2022	davekon7	Securing the health of our soils is crucial to secure food for us and future generations! Hope India and France will be role models in presenting sustainable ways of farming, securing a minimum of 3% organic content in soils worldwide.	Positive
06/05/2022	coopnews	#Agriculture #coops apex @COPACOGECA says #organic farming is important in context of #climatechange, #Covid19 and #Ukraineconflict and calls EU to help drive #digital innovation, in its response to European Parliament's #Organic Action Plan report	Positive
06/05/2022	ABVista	#FollowFriday to @agricology, a hub for exchanging practical information on agroecology to support a transition to more sustainable #agriculture. We fully support this movement through our products and services!	Positive
06/05/2022	GoAgroecology	#intercropped lentils and sulla are growing beneath durum wheat in Volterra sampling crops density and waiting for some rain in the weekend #agroecology #farming #agricolturatoscana #volterra #tuscanyhills #Toscana	Neutral
06/05/2022	OpenFoodNetUK	Agroecology is a way of thinking about food systems that are beyond necessarily the farm gate.""	Neutral
06/05/2022	Wervelvzw	#agroforestry has been shown to extend the grazing season by up to 15 weeks through improved carrying capacity of the #soil and better #grassland utilisation.This also has a significant positive impact on #ammonia emissions from housed stock.""	Positive

07/05/2022	Galbraith_Rural	A productive and versatile mixed organic farming unit with purpose-built abattoir. About 164.60 Ha in total#Farmforsale | #Landforsale | #Abattoir | #Farm | #Farm365 | #Livestockunit | #Galbraith	Neutral
...			
23/05/2022	FFC_Commission	More about #Agroecology	Neutral
23/05/2022	FPhosphates	Do you know that inorganic feed phosphates are legally valid for use in organic farming? #BiodiversityDay #BiodiversityDay2022 #WorldBiodiversityDay2022 #InternationalBiodiversityDay #organicfarming	Neutral
23/05/2022	esm_magazine	@ALDI_SUISSE has introduced a new private-label organic brand consisting of milk, dairy products, and eggs sourced from antibiotic-free dairy farming.#organic	Neutral
23/05/2022	RossMcNally_	How did wildlife possibly cope before we came along with cattle/sheep? Livestock farming, organic or not, typically has net-negative biodiversity impacts. As for open habitats and communities, I'd rather see these created by keystone wild herbivores behaving naturally.	Negative
23/05/2022	PSBaker10	Well yes, fossil fuels are turned into food by modern agriculture. But organic farming can't feed 8 billion people, especially if they want to eat meat.	Negative
23/05/2022	CountrysideNews	Organic farming or flower strips – which is better for bees? Research team including @uniGoettingen assess the efficiency of agri-environmental measures from different perspectives	Neutral
23/05/2022	Didara	This is insanity! Just as industrialised agriculture promised to feed us and never has this iteration will not either. We need a swift transition to agroecological food systems in the UK. Small farmers and localised food systems feeds the world #agroecology	Positive
23/05/2022	landcoalition	"We work with women to address domestic and economic violence. During the pandemic we helped women learn how to make soap from natural herbs and taught them organic farming. In order to protect the soil we don’t use pesticides or chemicals." - Jamela Jazi at	Neutral
23/05/2022	DallynLee	Organic covers going in today for a customer. Bit of classic rolling action for you too. #farming #organic #classictractor	Neutral
16/05/2022	FFC_Commission	We welcome Government's positive response to @CommonsEAC water quality report, and the commitment to Nature-based solutions like agroecology. Properly resourcing these recommendations and the delivery agencies is now the key to meeting these commitments.	Positive

VII. Código R del Caso 2

```
#CASO 2
#LA POLITICA AGRARIA COMUN
#English

##Setting directory
setwd("C:/Users/[redacted]")

##Instalar y cargar paquete para Academic Research
install.packages('academictwitteR')
library(academictwitteR)

#Establecer las credenciales de autorización
## Autenticación
set_bearer()
get_bearer()

#Búsqueda tweets
#Tweets del año 2017
tweets2017 <-
  get_all_tweets(
    query = "Common Agricultural Policy",
    start_tweets = "2017-01-01T00:00:00Z",
    end_tweets = "2017-12-31T00:00:00Z",
    file = "CAP2017",
    n = 25000,
  )

#Tweets del año 2018
tweets2018 <-
  get_all_tweets(
    query = "Common Agricultural Policy",
    start_tweets = "2018-01-01T00:00:00Z",
    end_tweets = "2018-12-31T00:00:00Z",
    file = "CAP2018",
    n = 25000,
  )

#Tweets del año 2019
tweets2019 <-
  get_all_tweets(
    query = "Common Agricultural Policy",
    start_tweets = "2019-01-01T00:00:00Z",
    end_tweets = "2019-12-31T00:00:00Z",
    file = "CAP2019",
    n = 25000,
  )

#Tweets del año 2020
tweets2020 <-
  get_all_tweets(
    query = "Common Agricultural Policy",
    start_tweets = "2020-01-01T00:00:00Z",
    end_tweets = "2020-12-31T00:00:00Z",
    file = "CAP2020",
    n = 25000,
  )
```

```r
#Tweets del año 2021
tweets2021 <-
  get_all_tweets(
    query = "Common Agricultural Policy",
    start_tweets = "2021-01-01T00:00:00Z",
    end_tweets = "2021-12-31T00:00:00Z",
    file = "CAP2021",
    n = 25000,
  )

#Tweets del año 2022
tweets2022 <-
  get_all_tweets(
    query = "Common Agricultural Policy",
    start_tweets = "2022-01-01T00:00:00Z",
    end_tweets = "2022-05-31T00:00:00Z",
    file = "CAP2022",
    n = 25000,
  )

#Sin RT
tweetsNORT <-
  get_all_tweets(
    query = "Common Agricultural Policy",
    start_tweets = "2017-01-01T00:00:00Z",
    end_tweets = "2022-05-31T00:00:00Z",
    file = "CAPNORT",
    is_retweet = FALSE,
    n = 25000,
  )

##Cargar paquete
library(httpuv)
library(rtweet)
library(ROAuth)
library(plyr)
library(dplyr)
library(stringr)
library(knitr)
library(tidytext)

#Quitar columnas que no nos interesan
tweets2017_tidy = tweets2017 %>% select(created_at,text,)
tweets2018_tidy = tweets2018 %>% select(created_at,text,)
tweets2019_tidy = tweets2019 %>% select(created_at,text,)
tweets2020_tidy = tweets2020 %>% select(created_at,text,)
tweets2021_tidy = tweets2021 %>% select(created_at,text,)
tweets2022_tidy = tweets2022 %>% select(created_at,text,)
NORTtweets_tidy = tweetsNORT %>% select(created_at,text,)

#Juntar los dataframes en uno solo
tidy1_as = as.data.frame(tweets2017_tidy)
tidy2_as = as.data.frame(tweets2018_tidy)
tidy3_as = as.data.frame(tweets2019_tidy)
tidy4_as = as.data.frame(tweets2020_tidy)
tidy5_as = as.data.frame(tweets2021_tidy)
tidy6_as = as.data.frame(tweets2022_tidy)
merge_CAP = merge(x=tidy1_as,y=tidy2_as,all = TRUE)
merge_CAP = merge(x=merge_CAP,y=tidy3_as,all = TRUE)
```

```
merge_CAP = merge(x=merge_CAP,y=tidy4_as,all = TRUE)
merge_CAP = merge(x=merge_CAP,y=tidy5_as,all = TRUE)
merge_CAP = merge(x=merge_CAP,y=tidy6_as,all = TRUE)

#Pasarlo a formato csv y guardarlo en el ordenador
write.csv2(merge_CAP,"CAPtweets")
write.csv2(NORTtweets_tidy,"CAPtweetsNORT")


# Se renombran las variables con nombres más prácticos
CAP_todos <- CAP_todos %>% rename(fecha = created_at,
                                  texto = text)
NORTtweets_tidy <- NORTtweets_tidy %>% rename(fecha = created_at,
                          texto = text)

#PAQUETES NECESARIOS
install.packages('quanteda')
install.packages('purrr')
library(quanteda)
library(purrr)

#LIMPIEZA
limpiar_tokenizar <- function(texto){
  # El orden de la limpieza no es arbitrario
  # Se convierte todo el texto a minúsculas
  nuevo_texto <- tolower(texto)
  # Eliminación de páginas web (palabras que empiezan por "http." seguidas
  # de cualquier cosa que no sea un espacio)
  nuevo_texto <- str_replace_all(nuevo_texto,"http\\S*", "")
  # Eliminación de signos de puntuación
  nuevo_texto <- str_replace_all(nuevo_texto,"[[:punct:]]", " ")
  # Eliminación de caracteres de control
  nuevo_texto <- str_replace_all(nuevo_texto,"[[:cntrl:]]", " ")
  # Eliminación de números
  nuevo_texto <- str_replace_all(nuevo_texto,"[[:digit:]]", " ")
  # Eliminación de espacios en blanco múltiples
  nuevo_texto <- str_replace_all(nuevo_texto,"[\\s]+", " ")
  # Eliminación de graficos
  nuevo_texto <- str_replace_all(nuevo_texto,"[^[:graph:]]", " ")
  return(nuevo_texto)
}

# Se aplica la función de limpieza a cada tuit
CAP_todos <- CAP_todos %>% mutate(texto_limpio = map(.x = texto,
                                                     .f = limpiar_tokenizar))
CAP_todos %>% select(texto_limpio) %>% head()
CAP_todos %>% slice(1) %>% select(texto_limpio) %>% pull()

#Distribución de tuits a lo largo del tiempo sin distinción de categoría
install.packages("lubridate")
library(lubridate)
library(ggplot2)
ggplot(CAP_todos, aes(x = as.Date(fecha))) +
  geom_histogram(position = "identity", bins = 20, show.legend = FALSE) +
  scale_x_date(date_labels = "%m-%Y", date_breaks = "6 month") +
  labs(x = "fecha de publicación", y = "número de tuits") +
  theme_bw() +
  theme(axis.text.x = element_text(angle = 90))
```

```
#Analisis del sentimiento
#Con la funcion score.sentiment empleada en el Caso 1
#Carga de lexicon
pos.words <- scan("C:/Users/[redacted]/POSITIVE_LEXICON_eng.txt",
what="character")
neg.words <- scan("C:/Users/[redacted]/NEGATIVE_LEXICON_eng.txt",
what="character")

#APLICAMOS LA FUNCIÓN EN EL DATAFRAME ANOTADO
PRUEBACAP01 <- score.sentiment(tweets = CAP_todos$texto_limpio,
pos.words,neg.words,.progress = 'text')

##Nueva variable con la categoría de sentimiento
PRUEBACAP01$categoria <- ifelse(PRUEBACAP01$score > 0, "Positive",
                                ifelse(PRUEBACAP01$score < 0, "Negative",
"Neutral"))

##Recuento por categorías
POSP <- length(which(PRUEBACAP01$categoria ==
"Positive"))/length(PRUEBACAP01$categoria)
NEUP <- length(which(PRUEBACAP01$categoria ==
"Neutral"))/length(PRUEBACAP01$categoria)
NEGP <- length(which(PRUEBACAP01$categoria ==
"Negative"))/length(PRUEBACAP01$categoria)

##Crear otro dataframe con las nuevas variables
WP_CAP <- data.frame(PRP = c(POSP, NEUP, NEGP), cat = c("Positive",
"Neutral", "Negative"))


#Gráfico circular
WP_CAP$cat <- factor(WP_CAP$cat,
                     levels = c("Positive", "Neutral", "Negative"))
##Sintaxis para la creación del gráfico
library('plotly')
WPP <- plot_ly(WP_CAP, labels = WP_CAP$cat, values = WP_CAP$PRP, type =
'pie',
                sort = FALSE,
                direction = "clockwise",
                textposition = 'inside',
                textinfo = 'label + percent',
                insidetextfont = list(color = '#FFFFF'),
                marker = list(colors = c('green', 'grey', 'pink'),
                              line = list(color = '#FFFFF', width = 0.25)))
%>%
  layout(title = 'CAP Twitter Sentiment Analysis', showlegend = TRUE,
         xaxis = list(showgrid = FALSE, zeroline = FALSE, 'showticklabels' =
FALSE),
         yaxis = list(showgrid = FALSE, zeroline = FALSE, 'showticklabels' =
FALSE))
WPP

#Juntar ambos dataframes en uno solo
tidy_as = as.data.frame(CAP_todos)
tidy2_as = as.data.frame(PRUEBACAP01)
juntos_tweets = merge(x=CAP_ todos, y=PRUEBACAP01, all = TRUE)

#Límite de memoria
#Guardamos ambas dataframes y los unimos en Excel para reducir su tamaño
```

```
write.csv2(tidy_as,"tweets_CAP_11")
write.csv2(tidy2_as,"tweets_CAP_21")


#Subimos el dataframe juntado en Excel para reducir tamaño
library(readxl)
ruta_excel3 <- "C:/Users/[redacted]/juntos_tweets.xlsx"
juntos_tweets <- read_excel(ruta_excel3)

#Distribución de tuits a lo largo del tiempo distinguiendo entre categorías
tweets_mes <- juntos_tweets %>% mutate(mes_anyo = format(fecha, "%Y-%m"))
tweets_mes  %>% group_by(categoria, mes_anyo) %>% summarise(n = n()) %>%
  ggplot(aes(x = mes_anyo, y = n, color = categoria)) +
  geom_line(aes(group = categoria)) +
  labs(title = "Número de tweets publicados", x = "fecha de publicación",
       y = "número de tweets") +
  theme_bw() +
  theme(axis.text.x = element_text(angle = 90, size = 6),
        legend.position = "bottom")

#Categorización de sentimientos solo en tuits originales
# Se renombran las variables con nombres más prácticos
tweetsNORT <- tweetsNORT %>% rename(fecha = created_at,
                                    texto = text)

#LIMPIEZA
# Se aplica la función de limpieza a cada tuit
tweets_CAP_NORT = data.frame(limpiar_tokenizar2(tweetsNORT$texto))

#APLICAMOS LA FUNCIÓN EN EL DATAFRAME SIN RT
CAPNORT2 = score.sentiment(tweets = tweets_CAP_NORT$texto_limpio,
pos.words,neg.words,.progress = 'text')
CAPNORT2$categoria <- ifelse(CAPNORT2$score > 0, "Positive",
                             ifelse(CAPNORT2$score < 0, "Negative",
"Neutral"))

##Recuento por categorías
POSP <- length(which(CAPNORT2$categoria ==
"Positive"))/length(CAPNORT2$categoria)
NEUP <- length(which(CAPNORT2$categoria ==
"Neutral"))/length(CAPNORT2$categoria)
NEGP <- length(which(CAPNORT2$categoria ==
"Negative"))/length(CAPNORT2$categoria)

##Crear otro dataframe con las nuevas variables
WP33_CAP <- data.frame(PRP = c(POSP, NEUP, NEGP), cat = c("Positive",
"Neutral", "Negative"))


#Gráfico circular
WP33_CAP$cat <- factor(WP33_CAP$cat,
                       levels = c("Positive", "Neutral", "Negative"))
##Gráfico
library('plotly')
WPPPP <- plot_ly(WP33_CAP, labels = WP33_CAP$cat, values = WP33_CAP$PRP, type
= 'pie',
                 sort = FALSE,
                 direction = "clockwise",
                 textposition = 'inside',
```

```
              textinfo = 'label + percent',
              insidetextfont = list(color = '#FFFFF'),
              marker = list(colors = c('green', 'grey', 'pink'),
                            line = list(color = '#FFFFF', width = 0.25)))
%>%
  layout(title = 'CAP Twitter Sentiment Analysis', showlegend = TRUE,
         xaxis = list(showgrid = FALSE, zeroline = FALSE, 'showticklabels' =
FALSE),
         yaxis = list(showgrid = FALSE, zeroline = FALSE, 'showticklabels' =
FALSE))
WPPPP

#PALABRAS USADAS CON MAYOR FRECUENCIA
#QUITAMOS ALGUNAS VARIABLES QUE NO NOS HACEN FALTA PARA SIMPLIFICAR
frec_words = tweets_mes_anyo_2 %>%
select(texto_limpio,categoria,categoria2,fecha)

#Primero ordeno las filas según negativo positivo o neutral y ahora extraigo
las filas de positivo y las negativas
df_ordenado <- frec_words[order(frec_words$categoria), ]

Negative_tweets = df_ordenado[1:14441,]
Positive_tweets = df_ordenado[14442:35230,]

#Frecuencia de palabras
library(tidyverse)
library(RColorBrewer)
library(wordcloud)
library(tm)
library(SnowballC)
library(RCurl)
library(XML)

Freq_Pos <- str_replace_all(Positive_tweets$texto_limpio, "@\\w+", "")
wordCorpus <- Corpus(VectorSource(Freq_Pos))
wordCorpus <- tm_map(wordCorpus, removePunctuation)
wordCorpus <- tm_map(wordCorpus, content_transformer(tolower))
wordCorpus <- tm_map(wordCorpus, removeWords, stopwords("english"))
wordCorpus <- tm_map(wordCorpus, removeWords, c("amp", "2yo", "3yo", "4yo"))
wordCorpus <- tm_map(wordCorpus, stripWhitespace)
wordCorpus <- tm_map(wordCorpus, removeWords,
c("common","agricultural","policy","will","can","cap","one","can","european",
"farming","good","great","want","saying","irenegarth", "may",
"farmers","mean","may", "agriculture", "euagri","lives"))

pal <- brewer.pal(9,"YlGnBu")
pal <- pal[-(1:4)]
set.seed (123)
wordcloud (words = wordCorpus, scale=c(5,0.1), min.freq = 5, max.words=50,
random.order=FALSE,
          rot.per = 0.35, use.r.layout=FALSE, colors=pal)

#Más cosas que se pueden hacer con la función wordcloud
wc2 = rquery.wordcloud(x=wordCorpus, type ="text",
                       lang = "english",
                       excludeWords = NULL,
                       textStemming = FALSE,
                       colorPalette = "Set1",
                       min.freq = 4,
```

```
                      max.words = 50)

tdm <- wc2$tdm
freqTable <- wc2$freqTable
head(freqTable, 10)

#Gráfico de barras con las 10 palabras más mencionadas
Barplot (freqTable[1:10,]$freq, las = 2,
        names.arg = freqTable[1:10,]$word,
        col ="lightblue", main ="Most frequent words",
        ylab = "Word frequencies")

#AHORA LO MISMO PARA LOS TUITS NEGATIVOS
Freq_Neg <- str_replace_all(Negative_tweets$texto_limpio, "@\\w+", "")
wordCorpus3 <- Corpus(VectorSource(Freq_Neg))
wordCorpus3 <- tm_map(wordCorpus3, removePunctuation)
wordCorpus3 <- tm_map(wordCorpus3, content_transformer(tolower))
wordCorpus3 <- tm_map(wordCorpus3, removeWords, stopwords("english"))
wordCorpus3 <- tm_map(wordCorpus3, removeWords, c("amp", "2yo", "3yo",
"4yo"))
wordCorpus3 <- tm_map(wordCorpus3, stripWhitespace)
wordCorpus3 <- tm_map(wordCorpus3, removeWords, c("farming","will","can",
"dont", "another", "common",
"agricultural","policy","sapere","cap","one","can","european","farming","want
","saying","irenegarth", "farmers","mean","may", "agriculture",
"euagri","lives","week","vivere","clima", "eau","biggest"))

wcneg = rquery.wordcloud(x=wordCorpus3, type = "text",
                         lang = "english",
                         excludeWords = NULL,
                         textStemming = FALSE,
                         colorPalette = "Set1",
                         min.freq = 2,
                         max.words = 50)

#Más cosas que se pueden hacer con la función wordcloud
tdmneg <- wcneg$tdm
freqTable2 <- wcneg$freqTable
head(freqTable2, 10)
#Gráfico de barras con las 10 palabras más mencionadas
barplot(freqTable2[1:10,]$freq, las = 2,
        names.arg = freqTable2[1:10,]$word,
        col ="red", main = "Most frequent words",
        ylab = "Word frequencies")
```

VIII. *Dataframe* del Caso 2

El *dataframe* al completo no se ha incluido por razones de espacio. Cualquier interesado puede solicitarlo a la autora.

FECHA	TEXTO	PUNTOS	CATEGORÍA
01/01/2017	@steviweavi ooh forgot the common agricultural policy.	**0**	**Neutral**
01/01/2017	RT @Energydesk: BREAKING: Investigation shows failures of subsidy system supposed to support UK farmers https://t.co/vkNbJ09qYE #moneyforno…	**-1**	**Negative**
01/01/2017	RT @Energydesk: Thanks @GeorgeMonbiot. Read the full EU farm subsidies investigation from @GPinvestigates here https://t.co/38adPzpoKx #mon…	**0**	**Neutral**
02/01/2017	#EU -support more projects keeping migrants at home. Investigate damage to #Africa from Common Agricultural Policy. https://t.co/s2z2NqvKz0	**0**	**Neutral**
02/01/2017	@Kishan_Devani "...suffer the full economic absurdities of the Common Agricultural Policy". 43 years and STILL the EU's CAP is 'absurd'.	**-2**	**Negative**
02/01/2017	RT @LeaveEUOfficial: Thanks to the EU’s Common Agricultural Policy, food is more expensive for Brits by 17% *IEA http://t.co/EgwRDUOWTy	**-1**	**Negative**
02/01/2017	RT @LeaveEUOfficial: Thanks to the EU’s Common Agricultural Policy, food is more expensive for Brits by 17% *IEA http://t.co/EgwRDUOWTy	**-1**	**Negative**
02/01/2017	RT @LeaveEUOfficial: Thanks to the EU’s Common Agricultural Policy, food is more expensive for Brits by 17% *IEA http://t.co/EgwRDUOWTy	**-1**	**Negative**
02/01/2017	RT @LeaveEUOfficial: Thanks to the EU’s Common Agricultural Policy, food is more expensive for Brits by 17% *IEA http://t.co/EgwRDUOWTy	**-1**	**Negative**
02/01/2017	RT @LeaveEUOfficial: Thanks to the EU’s Common Agricultural Policy, food is more expensive for Brits by 17% *IEA http://t.co/EgwRDUOWTy	**-1**	**Negative**
03/01/2017	After a 'post-truth' year Canada calmly leads with logical and scientifically informed food policy #GM #foodsecurity https://t.co/JjDERr5y6E https://t.co/f9YRL1QtzO	**2**	**Positive**
03/01/2017	Common Agricultural Policy: Treasury ups war on farm subsdies https://t.co/GAICTdK1jt	**0**	**Neutral**
	…		
28/05/2022	Stats for 1990. The UK provided 85% of its own food. What happened after that. The EU and the failed common agricultural policy. So not twaddle.<U+0001F923> https://t.co/WcCuZc5q20	**-1**	**Negative**
28/05/2022	RT @Lesliew16451240: Stats for 1990. The UK provided 85% of its own food. What happened after that. The EU and the failed common agricultur…	**-1**	**Negative**
29/05/2022	@amr_khafagy4 and @MauroVigani @CCRI_UK find a slow decline in factor-augmenting technical change in the EU agricultural sector. Common Agricultural Policy should	**0**	**Neutral**

	invest in agricultural innovations to improve productivity w/ sustainable practices. 10/13		
29/05/2022	Feeding Europe	**0**	**Neutral**
29/05/2022	RT @europeangreens: <U+0001F69C> We're discussing how a Common Agricultural Policy meeting the ambitions of the European Green Deal should look like w…	**1**	**Positive**
30/05/2022	@AliW_22 The Euro and the Common Agricultural Policy.	**0**	**Neutral**
30/05/2022	<U+0001F3A7>In this special edition of the EURACTIV agrifood podcast, @gerardofortuna & @NatashaFoote take you on a tour of 7 of Europe's capitals to hear about their Common Agricultural Policy (#CAP) strategic plans.	**0**	**Neutral**
30/05/2022	<U+0001F3A7>In this special edition of the EURACTIV agrifood podcast, @gerardofortuna & @NatashaFoote take you on a tour of 7 of Europe's capitals to hear about their Common Agricultural Policy (#CAP) strategic plans.	**0**	**Neutral**
30/05/2022	EU spending on climate action 'overstated' by €72 billion, mostly in common agricultural policy (#CAP), auditors say https://t.co/K40e0DEX6J	**-1**	**Negative**
30/05/2022	EUAgri: RT @EURACTIV: <U+0001F3A7>In this special edition of the EURACTIV agrifood podcast, @gerardofortuna & @NatashaFoote take you on a tour of 7 of Europe's capitals to hear about their Common Agricultural Policy (#CAP) strategic plans.	**0**	**Neutral**

IX. Limitaciones del análisis de opinión en RRSS

En el siguiente anexo, se describen en detalle las barreras que se han encontrado a lo largo de las distintas fases del proceso de analítica del sentimiento en redes (definición de requerimientos, determinación de la muestra, limpieza de datos y análisis).

A) Sesgos en la muestra: los usuarios de Twitter no son una muestra representativa de la población general pues, aunque la proporción de hombres y mujeres usuarios es similar, la distribución se inclina en gran medida hacia personas más jóvenes y mejor educadas que la media poblacional (Borrero & Zabalo, 2021). Por ello, cualquier muestra que se tome de esta plataforma no puede considerarse como una muestra representativa del conjunto de la población. Sin embargo, tal sesgo sistemático de la muestra puede disminuir con el tiempo a medida que más personas se conviertan en usuarios activos de Twitter.

Otro problema a la hora de definir los requerimientos de selección de la muestra radica en elegir correctamente la palabra o palabras clave que engloben de manera precisa el contenido que debe aparecer dentro de la muestra y que excluyan cualquier otra temática para así evitar incluir dentro otros *tuits* que no estén relacionados con el tema a analizar. Este problema no se puede reparar durante el pre-procesamiento de los datos y es debido a esto que la fase de selección de palabra/s clave es un paso tan importante. El analista debe hacerse una serie de preguntas críticas antes de la recopilación de datos (ver Casos 1 y 2), con implicaciones prácticas en la decisión de qué datos deben recopilarse (Stieglitz et al, 2018).

Por último, la falta de información también puede dar como resultado sesgos en el conjunto de datos. Como se ha visto en el Caso 1, el tamaño del conjunto de datos puede verse muy reducido si se desea filtrar, por ejemplo, en base a la geolocalización de los usuarios. Según Valkanas et al. (2014), solo el 1-2% de los *tuits* contienen coordenadas del Sistema de Posicionamiento Global (GPS). Ignorar todos los *tuits* sin etiquetas geográficas implica reducir enormemente el volumen del *dataset* de trabajo.

B) Calidad de la muestra: la calidad de la muestra con la que se trabaja al procesar datos de RRSS se puede ver comprometida por múltiples factores, entre ellos:

-El gran volumen de datos que se genera en tiempo real en estas plataformas continuamente y que dificultan su manejo (Stieglitz et al, 2018).

-Las conversaciones artificales provocadas por campañas específicas (a veces lanzadas desde usuarios falsos o *bots*), los anuncios publicitarios y el *spam*, así como los rumores y la información falsa (*fake news*), que pueden afectar negativamente en el comportamiento de otros usuarios de las RRSS (Stieglitz et al, 2018). Este tipo de *tuits* aumentan la cantidad de datos a procesar y dificultan los análisis. Concretamente, el *spam* y las noticias falsas, publicadas para respaldar o desacreditar cierto producto, pensamiento o personaje público, son muy complicadas de detectar y pueden tener un impacto grande en la percepción y las actitudes de los usuarios.

-La "suciedad" de los datos: muchos *tuits* incluyen *links*, imágenes, videos, audios o emoticonos que dificultan el pre-procesamiento de la muestra y NLP. Sin embargo, en muchas ocasiones estos elementos complementan el contenido del texto (por ejemplo, un *tuit* en el que el usuario muestra su disgusto con la PAC y en el que incluye varios emoticonos

de “enfado”, “rabia” o “tristeza”) y al eliminarlos se pierde información relevante de cara al análisis de sentimientos.

C) Limitaciones lingüísticas: asociadas al idioma particular del texto y al uso que hace el usuario del lenguaje en un contexto lingüístico determinado.

- Sarcasmo e ironía: a veces un texto puede contener frases de burla o emociones ocultas. Estas expresiones son difíciles de identificar y pueden dar lugar a resultados erróneos en el análisis de sentimientos (Patel & Patel, July 2020).

- Ambigüedad del lenguaje: la polaridad de una palabra puede cambiar según el contexto. Por ejemplo, la frase "La actitud que mostró durante la reunión fue negativa", refleja una mala opinión sobre el comportamiento de alguien, sin embargo también puede ser positivo en otro contexto muy diferente, por ejemplo, "El test covid salió negativo”. Por esta razón, un modelo que está entrenado para recopilar opiniones o un lexicon aplicado dentro de un dominio no pueden aplicarse a otros dominios o situaciones distintas. Lo ideal es contextualizar los métodos NLP con cada nuevo caso de estudio.

- Palabras gramaticalmente incorrectas: debido a la impulsividad, el ambiente informal y las restricciones en el uso de caracteres en Twitter, los usuarios no siguen correctamente las reglas gramaticales y ortográficas. Esto provoca que haya faltas de ortografía, abreviaturas, mayúsculas enfáticas, uso de jerga, etc. (Patel & Patel, July 2020).

- Contenido multilingüe: algunos *tuits* están escritos en varios idiomas, complicando aún más el proceso de *opinion mining*.

- Contradicciones: las personas pueden ser contradictorias en la forma en que revisan cualquier producto o expresan su opinión. Un mismo *tuit* puede reflejar dos puntos opuestos acerca de un mismo tema, haciendo imposible que el modelo pueda sacar una valoración del sentimiento agregado (global) que expresa dicho *tuit*.

- Jergas: las personas utilizan palabras o expresiones como "OMG", “DM”, "LOL" para expresar su respuesta. Identificar estas palabras e incluirlas en el lexicon o conjunto de entrenamiento requiere esfuerzos adicionales, especialmente si trabajamos con datos en otros idiomas o de otras partes del mundo, de las que no necesariamente se conoce el argot.

D) Precisión del método de aprendizaje no supervisado: se han identificado barreras inherentes a la metodología utilizada en este trabajo para la clasificación de sentimientos.

- Negación: el empleo de la negación altera la polaridad y el sentido de una palabra u oración. Uno de los mayores vacíos que presenta el método *lexicon-based* es su incapacidad para llevar a cabo una clasificación del sentimiento que tenga en cuenta el efecto de las estructuras lingüísticas de negación.

- Tamaño del *lexicon*: se necesitan una gran cantidad de recursos que alimenten el *lexicon* para obtener resultados altamente precisos (Patel & Patel, July 2020). La aparición de nuevas palabras y términos cada poco tiempo hace difícil mantener estos diccionarios actualizados al mismo ritmo al que evoluciona el lenguaje en las redes.

- El límite de caracteres: un *tuit* no puede contener más de 280 caracteres. Con esta limitación, es menos probable que dicho texto contenga una palabra coincidente con el diccionario o modelo de aprendizaje automático que se esté utilizando para el *opinion*

mining (Van Atteveldt et al, 2021). Por otro lado, en los textos breves también se ve restringida la cantidad de "pistas" contextuales disponibles (Barbieri et al, 2020) para que el modelo pueda extraer conclusiones en base a la materia planteada.

– El dominio: el análisis de opinión es una tarea condicionada en gran medida por el dominio (el ámbito) al que pertenece el conjunto de datos de prueba. Un *lexicon* adaptado a un campo específico o un modelo entrenado con un conjunto de datos de un determinado dominio (por ejemplo, opiniones sobre un equipo de fútbol), suele tener un rendimiento deficiente cuando se prueba en un conjunto de datos de otro dominio (por ejemplo, reseñas de restaurantes). El problema radica en que las palabras de opinión utilizadas para describir un evento en un dominio a menudo difieren de un dominio a otro (Patel & Patel, July 2020).

E) Temas legales y de privacidad de los usuarios: las normas reguladoras de las grandes RRSS en materia de protección de datos a menudo son muy restrictivas en lo que concierne a la compartición o el acceso a los datos de sus usuarios. Concretamente, Twitter es una de las empresas más abiertas a permitir el acceso a datos de sus usuarios a través de su API (ver Subapartado 3.1.). No obstante, conseguir extraer grandes volúmenes de *tuits* sigue estando restringido únicamente a usuarios de perfil académico, y cada solicitud de acceso al Academic Research Access es evaluada rigurosamente por el equipo del Twitter Developer Portal.

F) Limitaciones éticas: en línea con el punto anterior, cabe por último recordar que en trabajos de SMA se está tratando con datos personales de usuarios de RRSS, de los que muchas veces también se conoce su nombre de usuario, sexo y hasta su geolocalización. Es por ello que, pese a ser un enfoque de investigación emergente en el que aún no existe un criterio de actuación ampliamente aceptado, los expertos en SMA comienzan a plantear la necesidad de solicitar a los generadores de contenido su consentimiento para poder extraer y analizar sus publicaciones en RRSS. Una opinión expresada por los investigadores es que la información compartida en plataformas públicas sin contraseña o restricciones de membresía puede usarse para investigación sin necesidad de consentimiento informado (Salmons, 2017) y limita la obligación de solicitar ese consentimiento cuando se recopilan datos de plataformas o sitios web privados (como puede ser una cuenta privada de Instagram o Facebook). La otra perspectiva, sin embargo, es que siempre se debe procurar obtener el consentimiento informado de las personas cuyos datos están siendo analizados, aunque las opiniones difieren según el tema de estudio, el sitio web de donde se extrae dicha información y la población de muestra. Independientemente de la postura que se adopte sobre el consentimiento, obtenerlo de las personas puede, en la práctica, ser muy difícil (Salmons, 2017).