Enfoque interdisciplinar sobre el estudio cuantitativo de las emociones
Enfoque interdisciplinar sobre o estudo cuantitativo das emocións
An interdisciplinar approach to the quantitative study of emotions

Anastasiya Shevchenko

María del Carmen Rodríguez Rodríguez
Rubén Fernández Casal

Septiembre de 2019

Índice

Resumen

- *Resumen*

El análisis de las emociones clásicamente se ha desarrollado con técnicas de investigación cualitativas en sociología. En cambio, otras ramas del saber han ido abriendo paso hacia estudios automáticos y sistemáticos con la ayuda de diferentes lenguajes de programación.

El presente trabajo tiene por objeto revisar las herramientas cuantitativas disponibles para `R` con posibilidad de ser aplicadas a castellano. Para ello, se exponen las sucesivas actuaciones para la preparación de los datos de texto, su exploración y, por último, el cálculo de la polaridad emocional.

Finalmente, siguiendo el propósito de introducir estas técnicas en la sociología, así como llevar a cabo una aportación a una ciencia interdisciplinar de las emociones, se ofrece una serie de propuestas de mejora desde la computación, la estadística y la sociología.

Palabras clave: emociones, análisis cuantitativo, ciencia interdisciplinar, R, sociología.

- *Resumo*

A análise das emocións clásicamente desenvolveuse con técnicas de investigación cualitativas en sociología. En cambio, outras ramas do saber han ir abrindo paso cara a estudos automáticos e sistemáticos coa axuda de diferentes linguaxes de programación.

O presente traballo ten por obxecto revisar as ferramentas cuantitativas dispoñibles para `R` con posibilidade de ser aplicadas a castelán. Para iso, expóñense as sucesivas actuacións para a preparación dos datos de texto, a súa exploración e, para rematar, o cálculo da polaridad emocional.

Finalmente, seguindo o propósito de introducir estas técnicas na sociología, así como levar

a cabo unha achega a unha ciencia interdisciplinar das emocións, ofrécese unha serie de propostas de mellora desde a computación, a estatística e a sociología.

Palabras crave: emocións, análise cuantitativa, ciencia interdisciplinar, R, sociología.

- *Abstract*

The analysis of emotions has been classically developed on qualitative research techniques in sociology. Instead, other branches of knowledge have been making way for studies automatic and systematic with the help of different languages of programming.

The purpose of this paper is to review the tools Quantitative available for `R` with the possibility of being applied to Spanish. For this, the successive ones are exposed actions for the preparation of text data, its exploration and, finally, the calculation of emotional polarity.

Finally, following the purpose of introducing these techniques in sociology, as well as making a contribution to an interdisciplinary science of emotions, is made a series of proposals for improvement from computing, the statistics and sociology.

Keywords: emotions, quantitative, interdisciplinary analysis, R, sociology.

Agradecimientos

"Opinions are central to almost all human activities because they are key influencers of our behaviors." Bing Liu, Sentiment Analysis and Opinion Mining, 2012

La presente investigación surge a partir del esfuerzo y de la ilusión de aportar un grano de arena a los estudios sobre la esfera emocional de la realidad social, así como de intentar unir a investigadores de diferentes esferas, que tanto tienen que aportar a un bien común. En este sentido, este trabajo de Fin de Máster reune la energía, el tiempo y el afecto dedicado a él por todas aquellas personas que, de alguna forma, han colaborado en el proceso de su construcción.

Así, pues, considero que es relevante reservar un pequeño espacio para poder decir "Gracias".

Gracias a María del Carmen Rodríguez Rodríguez por ser mi guía, mi apoyo y mi idolo, por darme su compañía y su tiempo, por creer en mí y en mis decisiones susceptibles de ser puestas en duda.

Gracias a Rubén Fernández Casal por hacerme ver la importancia de la interacción entre las diferentes ciencias y por abrir ante mí este nuevo camino que estoy tomando.

Gracias a Raimundo Otero Enríquez por sus consejos sobre dudas existenciales, por estar siempre dispuesto a echar una mano y por recordarnos a todos nuestra propia valía.

Gracias a mi familia, por aguantar mi mal humor, a Xoán Antelo Castro, por dar la vuelta a mi vida, y, no menos importante, a todos aquellos que resuelven las dudas técnicas de los lenguajes de programación en foros abiertos.

Por último, gracias a José Antonio López Rey, que está *a punto de sentarse en el trono.*

Prólogo

"Knowledge in the matter of the Empire lies less in noting particular events than in studying a certain cast of mind", said the old diplomat.

Terry Pratchett, The Colour of Magic, 1983

El presente trabajo ha sido creado con el objetivo de analizar el *análisis de sentimientos* llevado a cabo con técnicas implementadas en el lenguaje de programación `R` (http://www.R-project.org/). La idea principal reside en, como señala Terry Pratchett, aproximarse a una forma de pensar diferente a la que los sociólogos estamos acostrumbrados, ya que estas herramientas han sido desarrolladas por informáticos y estadísticos, principalmente.

La investigación se estructura en tres bloques principales: el primer bloque incluye los primeros tres capítulos, donde se habla sobre el planteamiento general de la investigación, así como se explican los conceptos fundamentales del campo; el segundo bloque va desde el capítulo cuatro hasta el noveno y muestra la forma de proceder a la hora de preparar y analizar un conjunto de datos real y, por último, la tercera parte presenta una serie de propuestas de mejora de las técnicas estudiadas, así como las conclusiones generales.

Para llevar a cabo una introducción al mundo de `R` y aprovechar las diferentes posibilidades que ofrece, se tomó la decisión de escribir este trabajo en R-Markdown empleando el paquete [`bookdown`] (https://bookdown.org/yihui/bookdown/), lo que implica que el propio documento .pdf final que se ofrece ha sido programado. [1].

[1] Para elaborar este libro, se ha seguido principalmente como guía el siguiente documento: ["Escritura de libros con bookdown"](https://rubenfcasal.github.io/bookdown_intro


BY NC ND

Capítulo 1

INTRODUCCIÓN A LA PROBLEMÁTICA

El desarrollo de la Sociología de las Emociones, así como de las ciencias, en general, tradicionalmente viene ligado a un cierto rechazo hacia las emociones y una exaltación de la razón como único motor legítimo de la acción humana que podría considerarse adecuado. Sin embargo, hoy en día hay algunos científicos que tratan de desapegarse de esta premisa para abrir camino hacia una ciencia que tenga en cuenta las emociones como una parte inherente del ser: una ciencia 'con' emociones (Bericat, 2000).

Al desprecio hacia las emociones como parte de la vida social presente en todas las esferas, se suma la problemática de la desconexión que existe entre los distintos ámbitos del saber (Picard, 1995). Así, las temáticas de interés científico están llenas de sopalamientos y puntos de encuentro que no se aprovechan debido a las dificultades que producen las diferencias metodológicas y, sobre todo, las conceptuales.

Por ello, muchas aportaciones científicas en una misma dirección pasan desapercibidas al estar en mundos metodológica y epistemológicamente diferentes. R. W. Picard fue ignorado de este modo por los sociólogos, cuando en el año 1995, escribió su artículo sobre "la computación afectiva", que influenció fuertemente el posterior desarrollo de científicos dedicados al análisis de las emociones, así como la posterior inteligencia artificial.

En aquel artículo, el autor habla de la importancia de las emociones en la ciencia y la estigmatización a la que estas están sometidas por ser consideradas, *per se*, no científicas.

Esta gran aportación se basa, a su vez, en perspectivas procedentes del ámbito de las neurociencias, teniendo como referentes a Damasio (1994) y, por el lado de la psicología clínica, a Lewis (1995).

Siguiendo esta tendencia, el análisis cuantitativo de las emociones, se ha llevado a la práctica en los últimos años por parte de las grandes empresas en los últimos años, como Google, p ara conocer lo que opinan los consumidores de sus diferentes productos (Liu, 2012). La información utilizada se extrae a partir de, principalmente, "reviews" publicados en páginas web, consolidando así varias cuestiones clave: el interés aplicado al mercado de este tipo de análisis y la toma de teorías de referencia alejadas de la sociología (Pang and Lee, 2008).

Así, se ha desarrollado un cuerpo de análisis y un universo de significados alrededor de lo que son "los sentimientos", cómo se clasifican y cómo han de ser analizados, de forma que el resultado final suele ser una atribución de "opiniones" a una escala "positivo-negativo" (Basile, Basile, Nissim, Novielli, & Patti, 2017).

El presente documento presenta los resultados obtenidos a través de un proceso analítico inductivo, que nace a partir de la revisión de técnicas de análisis de sentimientos con R, pasa por la clarificación de las definiciones de conceptos clave ofrecidas por los desarrolladores y pensadores de estas técnicas, como por ejemplo, el propio 'análisis de sentimientos', para pasar a la puesta en marcha de la aplicación de algunas de las técnicas de este tipo de análisis existentes hoy en día para castellano. Resulta relevante la acotación lingüística debido al desarrollo casi exclusivo de las herramientas de minería de texto y del estudio de las emociones, en concreto, para la lengua inglesa. El efecto de este fenómeno se manifiesta en una atomización de la literatura hispana sobre la problemática y en la adaptación de las herramientas necesarias, como son los recursos léxicos, a través de traducciones automáticas del inglés . [1]

El propósito último consiste en llevar a cabo una reflexión acerca de la definición, la aplicación y el funcionamiento de las técnicas, así como las teorías de las emociones utilizadas como base (de las taxonomías, para plantear una serie de mejoras que contribuyan al desarrollo de la ciencia interdisciplinar de las emociones, así como al desarrollo de la sociología de las emociones y el análisis de sentimientos.

En este sentido, no se trata de investigar las emociones, sino partir de una serie de

[1] Un ejemplo de este tipo de léxico es AFINN, siendo una traducción automática de un diccionario inglés a múltiples idiomas. Una aproximación a su uso se puede encontrar en el siguiente enlace: https://rpubs.com/jboscomendoza/analisis_sentimientos_lexico_afinn

instrumentos y tratar de entrar en un universo de significados que mayoritariamente no ha sido formulado por científicos sociales. Esto es, buscar un punto de encuentro entre sociólogo/as, estadístico/as e informático/as.[2]

1.1 Objetivos

La finalidad de este trabajo consiste en llevar a cabo una revisión de las técnicas de análisis cuantitativo de las emociones en castellano desde teoría sociológica de la metodología. Se tratará, por lo tanto, de comprender sociológicamente la formulación y la aplicación de estas técnicas y se propondrá una serie de mejoras.

Por lo tanto, el objetivo general se puede enunciar de la siguiente forma:

- Revisar las técnicas de análisis cuantitativo de las emociones,'sentiment analysis', llevando a cabo pruebas sobre un conjunto de datos real y proponer una serie de mejoras.

En cuanto a los objetivos específicos, se lleva a cabo una agrupación en tres dimensiones, atendiendo a la afinidad de la naturaleza temática de las tareas que se incluyen en cada una. Así, se encuentran los objetivos de aproximación conceptual (teóricos), los objetivos empíricos y, por último, los objetivos aplicados a emociones concretas, en este caso, la emoción de la vergüenza.

Los objetivos de aproximación conceptual tienen la finalidad de estudiar y presentar las definiciones que se han dado a los diferentes términos del análisis de sentimientos por sus desarrolladores. Dentro de estos, encontramos los siguientes:

1. Presentar las definiciones de análisis de sentimientos y lenguaje de programación `R`.
2. Revisar las herramientas o paquetes disponibles en R para el análisis de sentimientos en castellano.

Los objetivos empíricos consisten en llevar a cabo pruebas de las funciones desarrolladas por los distintos paquetes de análisis de sentimientos para conocer su funcionamiento, así como las posibles deficiencias. Dentro de estos, encontramos los siguientes:

[2]En el presente trabajo se va a hablar únicamente del papel de las ciencias que se mezclan de forma abarcable para el estudio, por lo que no se hablará de la importancia de los lingüistas, a pesar de su crucial importancia. En este sentido, se trata de una acotación con fines de expresar únicamente aquellas esferas que se van a tratar específicamente.

5. Llevar a cabo pruebas de los paquetes disponibles, incluyendo:

    - Importación y lectura de los datos.
    - Preprocesamiento.
    - Conversión de datos de texto a variables numéricas con la `matriz de términos del documento`.
    - Descripción de los datos.
    - Formas de analizar, definir y detectar las emociones.
    - Aplicación de las puntuaciones de emoción obtenidas a técnicas de clasificación.

Por último, los objetivos aplicados consisten en analizar el papel de las emociones concretas dentro del análisis de sentimientos. Concretamente, en este caso, revisando la presencia de la emoción de la vergüenza en las técnicas de detección de las emociones y tratando de proponer una serie de actuaciones que pueden suponer un avance para el estudio de la esfera emocional en los datos de texto, de forma general. Por lo tanto, entre estos objetivos específicos se inscriben:

6. Estudiar la presencia de la emoción de la vergüenza en las técnicas de análisis de sentimientos.
7. Desarrollar una propuesta de mejora del análisis de las emociones de los datos de texto.

1.2 Aplicaciones

El análisis de sentimientos, como veremos con más detenimiento en los capítulos posteriores, pertenece a la esfera de la minería de texto, que, a su vez, se inscribe dentro del marco del procesamiento de lenguaje natural (Natural Language Processing, NLP). Así, se puede considerar como una herramienta que puede estar presente en todos los procesamientos de texto, si es considerada por los investigadores a la hora de la aplicación.

En este sentido, las principales aplicaciones de la minería de texto, según Rosas, Errecalde y Rosso (2010), son las siguientes:

- La extracción de información relevante de los datos de texto.
- La gestión de grandes volúmenes de información con técnicas de clasificación, resumen...

- La gestión del conocimiento, puesto que permite detectar rápidamente las claves de la información contenida.
- La aplicación del análisis de sentimientos.

Por lo tanto, el análisis de sentimientos se puede considerar como una herramienta por sí sola o más bien el apoyo o la ayuda para el desarrollo de otras. Si consideramos que se trata de una herramienta con entidad propia, las aplicaciones planteadas, siguiendo a Pang y Lee (2008), se pueden resumir en las siguientes:

- Análisis de reviews de páginas web para identificar las opiniones de los compradores sobre la calidad de los productos, así como la positividad o negatividad de sus experiencias. En este caso, el análisis de sentimientos se considera como una herramienta que alcanza el objetivo último por sí sola. Las técnicas más utilizadas en este enfoque son los clasificadores automáticos.

- Análisis de sentimientos como una herramienta para el desarrollo o la mejora de otro tipo de tecnologías, como pueden ser sistemas de recomendaciones o la identificación de comportamiento agresivo ('flame') en determinados entornos online. Para estos propósitos se suele utilizar la elaboración de resúmenes ('summarization').

- Análisis de sentimientos en estudios de negocio para identificar la realidad subjetiva creada alrededor de la imagen de la empresa, así como la de sus competidores. En estos casos las técnicas tradicionales de recogida de datos, como la encuesta, no serían capaces de recoger la subjetividad de las asociaciones al tratarse de una serie de datos no estructurada.

- Análisis de sentimientos aplicado a la Inteligencia gubernamental como una forma de identificar las fuentes temáticas o los entornos donde se pueden producir comunicaciones con contenidos hostiles, como podrían ser actividades terroristas.

- Análisis de sentimientos unido a diferentes áreas científicas. En esta parte se especifica que resulta útil tanto para el análisis político, como en otros ámbitos, por ejemplo, el de la sociología. En esta aplicación es fundamental el estudio de la polaridad de las relaciones o la negatividad o positividad de la interacción detectada entre diferentes actores.

Por ahora, hemos considerado únicamente las utilidades que puede aportar el análisis de sentimientos por sí solo o como parte de la minería de texto en un sentido que podríamos

considerar clásico. Es decir, una vez que se tiene claro que estas técnicas son aplicables y aplicadas al mundo de la empresa y/o bien de la inteligencia, sea gubernamental o de seguridad informática y semejantes, podemos preguntarnos: ¿y para qué más nos puede servir?

La detección y el análisis estadísticamente-sociológico de las emociones, de forma más o menos automática, pueden dar un paso más allá de la mayoría de las aplicaciones que se llevan a cabo actualmente, es decir, del análisis de mercados y l de las campañas de captación personalizadas de clientes online (Pang y Lee, 2008).

En este sentido, las aplicaciones del análisis de sentimientos se pueden trasladar a múltiples campos, si consideramos este área como un apoyo para el avance de otras ciencias:

- Contribución a la eficiencia del procesamiento de los datos cualitativos a partir de las entrevistas en profundidad, historias de vida... con el fin de extraer conclusiones a partir de un análisis no memorístico.

- Detección de las temáticas más relevantes o con una carga emocional de unas determinadas características para plantear un acercamiento a determinados grupos de interés sensibles o en situación de riesgo.

- Posibilidad de un planteamiento de teorías basadas en la experimentación con datos reales de dinámicas y estructuras afectivas, donde se implica más de una emoción de forma más o menos reinterada y sistemática (Bericat, 2016).

- Detección de colectivos en riesgo de algún tipo de violencia a partir de datos obtenidos en entornos online, como Facebook u otras redes sociales.

- Posibilidad de analizar realidades sociales complejas sin llevar a cabo recogida de datos con trabajo de campo, además de las ventajas de procesamiento posterior.

Además de estas aplicaciones, podría proponerse un amplio abanico de posibilidades para su aplicación a la hora de entrar en otros ámbitos, como podría ser el de la docencia. Esto es, estas herramientas podrían servir de ayuda para los docentes a la hora de actualizar sus conocimientos o preparar temarios identificando las temáticas más actuales o bien cambios significativos en aspectos concretos que se enseñan. Del mismo modo, ser capaces de enseñar a los alumnos a trabajar procesando cantidades considerablemente grandes de texto permite que todos seamos capaces de ahorrar tiempo y esfuerzo a la hora de tomar

decisiones sobre qué leer, qué estudiar, etc.

Capítulo 2

CONSIDERACIONES METODOLÓGICAS

El planteamiento metodológico corresponde a un estudio de exploración de las técnicas de análisis de sentimientos en cuanto a su definición, su funcionamiento y aplicación a las distintas áreas del conocimiento y de la investigación. Así, para cumplir los objetivos no resulta suficiente con un único enfoque, por lo que se plantea una metodología múltiple: un estudio documental, un estudio cuantitativo y el desarrollo de una discusión sobre lo analizado, incluyendo una reflexión sobre cómo mejorar las técnicas.

2.1 Metodología documental

Para cumplir con los primeros cuatro objetivos contextuales de la investigación, se lleva a cabo una revisión global de las técnicas de análisis de sentimientos a partir de los repositorios oficiales, como Github y Cran.r-project. A partir de esta, se procede a acotar los elementos seleccionados que admiten el idioma castellano para el análisis de texto.

Siguiendo este procedimiento, se obtienen las guías disponibles online para cada uno de los paquetes, redactadas como guias de ayuda donde se definen las funciones que incluye el paquete y otros paquetes afines necesarios para su ejecución. Una vez analizadas *grosso modo* las operaciones que aporta cada una de las herramientas, se procede a determinar cuáles son las taxonomías de las emociones utilizadas. Es decir, no se trata de un procedimiento

bibliográfico a partir del cual se determina qué emociones deberían ser consideradas, sino que se estudia cuáles han sido utilizadas en las herramientas en cuestión.

Finalmente, se revisan los documentos que contienen los 'diccionarios' o el conjunto de palabras referidas a las emociones que la técnica será capaz de detectar. La importancia de este punto reside en la formulación concreta de cada diccionario porque, aunque la puntuación de las palabras está habitualmente planteada en grados de positivo, negativo y neutro, los resultados pueden no ser comparables si la escala de valoración no es la misma. Es decir, la diferencia en la formulación de la unidad de medida de los diccionarios supone que no sean combinables entre sí.

2.2 Metodología cuantitativa

La metodología cuantitativa, formulada como el procesamiento y los análisis estadísticos de los datos de texto, constituye la parte central del tipo puesto que permite extraer conclusiones sobre las técnicas existentes en el análisis de sentimientos en R, acotado a lo que se puede aplicar a castellano. Por lo tanto, se trata de un procedimiento de análisis de las técnicas a través de la presentación de unos datos concretos, en este caso, la base de datos CorpusCriticasCine [1].

En las siguientes páginas, veremos se plantearán los principales problemas de la aplicación de técnicas cuantitativas de procesamiento de datos extraídos de un entorno digital, así como la influencia que han ejercido sobre la toma de decisiones a lo largo del desarrollo del proceso investigador. Esta exposición se distribuye en dos partes: la exposición de las dificultades inherentes en parte, esperadas *a priori* y, por otro lado, los cambios producidos a lo largo de la definición de la problemática de estudio.

2.2.1 Dificultades inherentes

El procesamiento de texto y la metodología aplicada a los datos extraídos del mundo digital contienen, de forma inherente, una serie de dificultades a priori, que se han de plantear y considerar en toda investigación. En este sentido, siguiendo el planteamiento de Castellanos,

[1] El archivo que contiene la base de datos, así como el código completo del presente trabajo en su estado anterior a la compilación en formato .Rmd se encuentra disponible en el siguiente enlace: https://www.dropbox.com/sh/i0qm1ao7izxm69c/AADVgvyBmYI4jDDMgj8dh9Kla?dl=0

Gómez Yáñez y Moraño (2019), resultan relevantes los siguientes puntos, planteados por ellos como retos:

Reto 1:

El tratamiento de datos voluminosos se constituye como un gran reto para las técnicas estadísticas y metodológicas, en general, aplicadas al procesamiento de los datos. En este sentido, el presente trabajo trata de mostrar la utilidad y la eficiencia del procesamiento automático de texto para un conjunto de textos consideradamente reducido (3787 críticas de cine). El volumen de datos procesado en la parte práctica, de ser tratado cualitativamente, sería de una naturaleza abarcable únicamente si el tiempo disponible y la financiación fuesen amplios.

Reto 2:

La representatividad estadística es otro criterio que se suele considerar prácticamente imprescindible en las investigaciones aplicadas no se refiere a esta metodología concreta porque no se conoce a la muestra, ni lo datos se han obtenido llevando a cabo trabajo de campo. Para solucionar este problema, se plantearía una serie de ajustes metodológicos en la recogida u obtención de datos.

Por otro lado, las técnicas de análisis de sentimientos están encaminadas a analizar un alto volumen de datos, por lo que no sería necesario muestrear los datos, además de tener por objetivo conocer los datos *per se*. Esto es, no conocemos a los individuos y lo que queremos saber de ellos es lo que sienten en relación al contexto y a la expresión que han utilizado. Siempre y cuando no conozcamos las características de perfil de las unidades de análisis, el objetivo de extrapolar no es planteable.

Reto 3:

En el ámbito sociológico, más que en el estadístico y, por supuesto, que en el computacional, el investigador depende de la herramienta a la hora de analizar. La plataforma de la que se extrae la información, por un lado, decide los datos que podremos extraer, y, por otro lado, la herramienta de procesamiento de texto (para el caso del presente estudio), decide

los análisis que les podremos aplicar y las conclusiones que podremos extraer. Esto es, el análisis se encuentra condicionado de forma previa a las decisiones del propio investigador.

En este sentido, se produce una doble vertiente de limitaciones: por un lado, de las técnicas y, por el otro lado, las limitaciones éticas y legales, que se han de tener en cuenta en cuanto a las políticas de protección de datos, así como de la anonimización de los mismos. En este trabajo se analizan y se cuestionan las limitaciones de naturaleza técnica para los investigadores de las ciencias sociales, concretamente de los y las sociólogos/as.

Reto 4:

Asimismo, por el lado de la especialización, plantearemos y reforzaremos la realidad de la necesidad de contribuir al desarrollo de las herramientas de análisis de datos digitales de cualquier naturaleza, y, por lo tanto, lo imprescindible de introducirse en lo digital, cuyos avances generalmente se hacen ajenos.

Así, los dos últimos retos que se plantean estos autores transmiten la esencia del presente trabajo en la era de un procesamiento de datos que ya no es completamente relegado al trabajo manual, sea de análisis o de codificación previa. Se trata de las limitaciones metodológicas y la especialización.

2.2.2 Formulación cambiante del proceso investigador

Como en todo proceso de investigación, a lo largo de su desarrollo los objetivos y la propia esencia del trabajo se ha visto reformulada. Si bien la naturaleza de los datos por obtención los sitúa en todo caso como datos sencundarios, en el sentido de no ser obtenidos directamente por trabajo de campo, la idea inicial de estudiar la emoción de la vergüenza desde la perspectiva del análisis de sentimientos ha sido desechada en seguida.

En las técnicas de análisis de sentimientos existe un problema de formulación esencial: las emociones propiamente se suelen considerar como no relevantes, frente a la positividad o negatividad que el individuo expresa en relación al objeto (‘target’) sobre el que recae esa emoción. En ese sentido, finalmente la investigación tomo una redirección hacia el análisis de las técnicas de forma general, teniendo en cuenta la anterior problemática teórico-conceptual.

Sería interesante ser capaces de clasificar los textos según las emociones expresadas o, de forma simplificada, según la emoción que se exprese predominantemente. En cambio, este tipo de planteamientos no ha sido posible por dos razones: la ausencia de datos supervisados y la ausencia de procedimientos que especifiquen las emociones.

En primer lugar, los datos supervisados (Mur) son aquellos que han sido procesados y clasificados previamente por algún investigador. Estos son de gran interés porque permiten que se pueda comprobar la bondad de la clasificación o, dicho de otro modo, en qué medida acierta el clasificador. El problema con el que nos encontramos es que no se ha logrado localizar un conjunto de datos de estas características que, además, esté en castellano y, por otro lado, clasificado por emoción y no por temas.

Por lo tanto, se ha tomado un conjunto de datos no supervisado en castellano que contiene una serie de críticas de cine. Este conjunto de datos, como veremos posteriormente, se distribuirá en variables, dentro de las cuales se encuentran las dos variables de mayor interés para el presente trabajo: el resumen de la opinión o sumario y el texto.

Capítulo 3

EL ESTADO DE LA CUESTIÓN

El presente trabajo tiene una naturaleza inherentemente aplicada y enfocada hacia una metodología concreta. Por este motivo, la fundamentación teórica que se expondrá no busca ser exhaustiva, sino trata de encontrar la precisión conceptual y terminológica de las técnicas de *sentiment analysis*.

Este territorio ha permanecido en gran medida sin explorar por las ciencias sociales, así como por el ámbito académico en general, exceptuando la rama de la computación lingüística. Así, antes de ser capaces de formular cualquier conclusión o proponer actuaciones, es preciso hacer una exploración, tratando de comprender cómo ha sido pensado por sus creadores.

En lo que se refiere a las consideraciones teóricas, se lleva a cabo una aproximación introductoria hacia los principales conceptos relacionados con el lenguaje de programación `R`, el análisis de sentimientos y la terminología utilizada en él.

3.1 Una breve definición

El análisis de sentimientos es un conjunto de técnicas pertenecientes al procesamiento automático del lenguaje natural, es decir, expresado en texto (Vilares, 2017). Esta esfera se diferencia dentro del análisis de texto por su objetivo consistente en estudiar y extraer conclusiones acerca de las realidades subjetivas presentes en el discurso. Esta rama procede de lo que se conoce como `NLP` o procesamiento de lenguaje natural, ligado a la minería de texto y minería web. En la definición de análisis de sentimientos, se expone que "analiza

las opiniones, sentimientos, evaluaciones, actitudes y emociones de los individuos a partir del lenguaje escrito" (Liu, 2012).

3.1.1 Tradiciones según la naturaleza de los datos

Existen principalmente dos tradiciones de análisis de sentimientos, diferenciados por la naturaleza de los datos y el planteamiento de las técnicas, teniendo cada una de ellas objetivos, por consiguiente, diferenciados. Los datos, por un lado, pueden ser no supervisados, en los cuales no hay ninguna variable que nos puede dar información acerca de si las conclusiones que estamos extrayendo son acertadas o en qué medida el instrumento de medida funciona adecuadamente, o bien supervisados, si han sido tratados por expertos y clasificados, por lo que se dispone de una información para constrartar la bondad de las técnicas aplicadas.

Estas dos tradiciones o estrategias son principalmente los siguientes (Vilares, 2017):

3.1.1.1 Estrategia basada en el conocimiento

La estrategia basada en el conocimiento, *knowledge based strategy*, se caracteriza por el trato con datos de naturaleza no supervisada y el uso de diccionarios de palabras puntuadas en función de la positividad o negatividad de la emoción o la carga emocional. El software habitualmente utilizado en este enfoque es R. Este conjunto de procedimientos tiene, a su vez, dos enfoques principales:

3.1.1.1.1 Análisis semántico

Estudio léxico de los datos no supervisados con la ayuda de diccionarios, persiguiendo la finalidad de describir los datos o bien poner a prueba diferentes técnicas (como en el caso del presente estudio). Los diccionarios son conjuntos de términos, cada uno de los cuales tiene asignada una determinada puntuación, dependiendo de si se considera de naturaleza positiva o negativa. En este sentido, solamente se van a detectar y categorizar aquellas palabras que aparezcan en estos diccionarios y de forma en la que están definidas sus posibilidades de aparición.

La utilización de los diccionarios es el enfoque más próximo a las ciencias sociales, en el que se puede plantear un léxico específico para diferentes problemas de estudio. Esto incluye la

identificación de las emociones, cuestión que no ha recibido excesiva atención a lo largo del desarrollo de *sentiment analysis*.

3.1.1.1.2 Análisis sintáctico

Estudio sintáctico para datos en un solo idioma, contemplando la posibilidad de establecer una combinación de las dos técnicas para los datos no supervisados.

3.1.2 Estrategia del aprendizaje automático

La estrategia basada en el aprendizaje automático, *machine learning strategy*, se caracteriza por el trato con datos supervisados, lo que permite estudiar la bondad de ajuste, y la aplicación de algoritmos con fines de clasificación, predicción, etc. Esta rama contiene una diferenciación interna atendiendo a su aplicación a conjuntos de datos de un solo idioma o varios al mismo tiempo. El software utilizado habitualmente para este tipo de análisis es Python.

En este caso se suelen aplicar las redes neuronales o SVM, máquinas de soporte vectorial. Las redes neuronales son modelos computaciones constituidas por 'nodos' interconectados entre sí, que tienen el fin de, a partir de las conexiones, ser capaces de clasificar o predecir unos determinados datos. Por otro lado, las máquinas de soporte vectorial son transformaciones no lineales realizadas a los datos en relación a un espacio (Benacourt, 2005). Estas técnicas presentan un gran inconveniente: sus resultados son difícilmente interpretables.

3.2 Emociones y terminología

Teniendo en cuenta que el lenguaje es lo que hace posible la comunicación y la comprensión, la terminología utilizada por los desarrolladores de las técnicas de análisis de sentimientos es clave a la hora de entender cómo están constituidas las técnicas y, sobre todo, qué es exactamente lo que se estudia.

Revisando principalmente a Bing Liu (2015) como el máximo exponente de la literatura especializada en el campo del análisis de sentimientos, resulta apreciable la consideración que se le da a los diferentes problemas inherentes del objeto de estudio, derivados de una cuestión central: la expresión de la subjetividad de las personas.

Así, se tienen en cuenta numerosos problemas, como por ejemplo, la diferencia entre la emoción expresada explícita e implícitamente, la ironía y el sarcasmo y otras tantas problemáticas propias de estas técnicas y de la complejidad de su desarrollo.

Un inconveniente clave surge en el momento en el que se define la palabra *sentimiento* como un sinónimo de *opinión* [1]. Dentro de este concepto de opinión se diferencian los siguientes niveles:

- Opinión comparativa y regular, pudiendo ser esta última directa o indirecta.
- Opinión subjetiva y basada en hechos.

En cuanto a las emociones, que nunca son denominadas como tales, son procedentes de la psicología, siguiendo a autores como Parrott (2001), Arnold (1960)... En todo caso, el principal acento y desarrollo de las herramientas está siempre planteado en función de la polaridad: lo positivo, negativo o neutro de una *opinión*.

Por lo tanto, a pesar de lo atractivo que suena, una vez entrados a estudiar este ámbito, nos damos cuenta de que el planteamiento que se encuentra en el corazón de esta técnica se basa en la asignación de una puntuación positiva o negativa de la emoción del individuo hacia un objeto o "target" concreto (Almashraee, Monett, and Paschke, 2016) y no en la identificación de la emoción o sus relaciones con los argumentos descritos.

Esto revela el objetivo de la creación de las técnicas: determinar si los individuos se predisponen positiva o negativamente hacia algo. Por lo tanto, las emociones realmente no importan. Lo que sí importa es saber si uno se siente bien o mal en relación a ese objetivo de la cámara Canon y, por consiguiente, si la comprará o recomendará a sus amigos.

Viendo esto, se esclarece que el desarrollo de las herramientas no está preparado, ni planteado, para el estudio de las emociones, lo que hace que se pierda la mayor parte del potencial que se espera al escuchar de su existencia.

[1] Las expresiones terminológicas suponen una problemática que no se estudiará exhaustivamente en el presente trabajo. En cambio, se ha de indicar que no se puede pasar por alto la propia relevancia de la esfera emocional denominada como *sentimiento*

3.3 Procesamiento de texto y niveles de análisis

Aunque el análisis de sentimientos sea una rama específica dentro del análisis de texto, esto no evita la necesidad de su procesamiento, tal como una forma de preparación para la aplicación de las técnicas. Esto es, se ha de llevar a cabo un análisis de texto, sin atender a la esfera emocional primero, para poder establecer posteriormente determinados objetivos relacionados con esta.

Así pues, dependiendo de los aspectos que se quieren analizar concretamente, el preprocesamiento de los datos varía enormemente. En este sentido, el primer paso conlleva un manejo de los diferentes formatos en los que los datos de texto son contenidos, dentro de los cuales podemos encontrar .DOC, .PDF, .XML, .HTML, JSON, etc.

Además de tener en cuenta el contacto inicial y la preparación de una estructura de texto accesible y procesable, se crea un objeto denominado `corpus`, que establece una forma específica de organización interna de los datos, así como una determinada información.

En el procesamiento del lenguaje natural se plantean principalmente tres tipos de análisis según el nivel del contenido y la estructura que se desea estudiar (Pozzi, Fersini, Messina y Liu, 2017):

- El nivel de *mensaje* determina cuál es la puntuación de polaridad de la opinión que se presenta en un texto.
- El nivel de la *oración* supone que en cada una de ellas se encuentra una opinión específica.
- El nivel de *entidad* o de *aspecto* contempla la importancia de una opinión teniendo en cuenta el objeto o *target* al que va dirigida, por lo que combina el nivel del mensaje y el de la oración, identificando las opiniones en relación a cada *target*.

En este caso, siendo la formulación con el programa `R`, se utiliza el paquete `tm`, escogiendo como entidad propia las palabras, dentro de los diferentes niveles que se pueden tener en cuenta en de la herramienta. Partiendo de las palabras:

- El texto se puede descomponer en *caracteres*, cada una de las cuales sería una unidad independiente e influyente en el resto. Por caracter se entienden no solamente las letras, sino también los números, los espacios, etc.

- Se puede considerar como nivel principal la *palabra* en un sentido semántico. En este

tipo de planteamiento del análisis se suelen llevar a cabo procedimientos de reducción de la variabilidad, con el fin de reducir el volumen de datos, puesto que cada palabra equivaldría en algunas técnicas a una variable.

- Por último, dentro del texto se puede conceder la importancia a los *conceptos*. La diferencia principal del nivel de la palabra y el concepto, está en que este último tiene en cuenta la sinonimia y la polisemia.

3.4 Uso del lenguaje de programación `R`

`R` es un lenguaje de programación (R Core Team), gratuito y de código abierto, con una comunidad activa, destinado y utilizado principalmente con fines estadísticos (Paradis & Ahumada, 2003). Esta herramienta, como es propio de los lenguajes de programación, se constituye como un espacio de creación, por lo que las posibilidades de desarrollo, así como de aplicación son numerosas.

Este lenguaje se descarga con una interfaz gráfica mínima, que permite incluir "paquetes" o librerías que faciliten el tratamiento sin tener que utilizar el código o la sintaxis con conocimientos de desarrollador. Es decir, determinadas herramientas están formuladas en *funciones* o *comandos* [2], que vamos a utilizar para llevar a cabo nuestros análisis (Oliden, 2009).

Cuando se descarga `R`, la instalación incluye un conjunto de instrucciones básicas (paquete `base`), a partir de las cuales se pueden llevar a cabo análisis o bien procesar los datos. Si queremos utilizar herramientas especializadas, será necesario instalar 'paquetes' o, dicho de otro modo, conjuntos de funciones que hayan sido previamente desarrolladas, a no ser que seamos capaces de crearlas nosotros mismos.

Así, se puede decir que está creado "por partes": hay un núcleo con una serie de instrucciones `base` a las que posteriormente se van añadiendo "colecciones de comandos", los denominados "paquetes". Estos, por su lado, son desarrollados para que no haya que hacer manualmente determinadas utilidades programándolas y pudiendo usar una sintaxis sencilla, automatizando y simplificando así muchas las utilidades o tareas que se pueden querer

[2] Las funciones son las instrucciones concretas que se dan al lenguaje de programación para que se lleve a cabo cualquier acción deseada. Es posible formular varias acciones en una misma función, lo que facilita significativamente el proceso de programación y de utilización del lenguaje en general.

llevar a cabo, y para no tener que descargar todas las utilidades a la vez, lo que haría la instalación inviable computacionalmente.

Además, se ha de tener en cuenta que la mayor parte de la metodología implementada se basa en contribuciones de colaboradores, lo que produce que haya varios paquetes para el mismo objetivo y su calidad no esté garantizada.

Por lo tanto, los principales componentes que se deben conocer son los comandos, que incluyen una única instrucción, las funciones, que pueden contener un conjunto amplio de instrucciones que se ejecuten de forma sucesiva o en bucle, y, por último, paquetes, que son conjuntos de funciones preparados para cumplir unos fines concretos.

Por otro lado, existe una serie de herramientas creadas con interfaz gráfica, como, por ejemplo, el paquete `rattle` para minería de datos numéricos. En cambio, esta formulación limita enormemente las opciones que se pueden aplicar. Por este motivo, se suele utilizar `R` desde la propia consola, sin interfaz.

3.4.1 Herramientas para análisis de sentimientos

Como comentábamos anteriormente, los paquetes disponibles para `R` están creados para cumplir con determinadas necesidades, por lo que también se encuentran disponibles librerías especializadas para el análisis de sentimientos. Entre estos se encuentran: `SentimentAnalysis`, `sentimentr`, `sentometrics`, `syuzhet` y `qdap`.

Sin embargo, el único paquete que finalmente ha sido posible utilizar para la elaboración de este trabajo es `qdap`. Esto se debe a un descarte sucesivo de los restantes paquetes, por las razones siguientes;

- `SentimentAnalysis` y `sentimentr`: no acepta idiomas, más allá de inglés.
- `sentometrics`: estudia series de tiempo para datos de texto.

Otra razón, que se constituye como la principal para descartar los paquetes existentes, es que ninguno de los paquetes, incluyendo `syuzhet`, documenta explícitamente con algoritmos y fórmulas los procedimientos que se aplican, por lo que los resultados del análisis no son interpretables desde nuestra perspectiva. Esto es, no hay transparencia en la definición de la metodología, aunque sea posible ver el código original, tratándose de un lenguaje abierto.

En lo que se refiere al paquete `qdap`, únicamente la función `polarity` ofrece una explicación

del procedimiento en base a una formulación matemática expresada en su guía disponible online [3]. A pesar de estos inconvenientes, esta será la única aplicación específica del análisis de sentimientos que será posible utilizar, sin tener en cuenta la preparación de los datos.

En cuanto a los diccionarios, algunos ejemplos que están traducidos a varios idiomas son el léxico *afinn* desarrollado por Finn rup Nielsen, *bing* desarrollado por Minqing Hu y Bing Liu y, por último, *nrc* creado por Mohammad, Saif M. y Turney, Peter D. Estos diccionarios, tal y como están formulados, no pueden combinarse directamente entre sí, puesto que tienen niveles de medición diferentes.

Por último, para llevar a cabo el análisis de sentimientos es preciso anteriormente procesar el texto, como veremos posteriormente. Este procesamiento se lleva a cabo, sobre todo, con la ayuda del paquete de minería de texto `tm`, que, a su vez, activa el paquete `NLP`.

3.4.2 Análisis de sentimientos para castellano

Como mencionábamos anteriormente, disponemos de herramientas creadas para la lengua inglesa, que con un pequeño "plus", que son los léxicos o diccionarios, pueden ser adaptables. Esto es, se podría decir que no hay paquetes elaborados con el fin de analizar emociones pensando en texto en castellano (Miranda & Guzmán, 2017).

Asimismo, habitualmente estos diccionarios, para idiomas que no sean inglés, suelen ser traducciones, más o menos automáticas, de diccionarios previamente elaborados para la lengua inglesa (Plaza del Arco & Jiménez-Zafra, 2018). Esto supone un relevante problema, en cuanto a que, por ejemplo, "disgusto" en castellano no tiene nada que ver con "disgust" en inglés, aunque, en la práctica, muchas veces se traten como si fuesen sinónimos.

En resumen, la práctica habitual a la hora de llevar a cabo el análisis de sentimientos es coger una herramienta hecha por y para inglés, traducirla a otro idioma, en este caso castellano, y analizar los resultados en base a la carga positiva o negativa del discurso a partir de aquellas emociones que se han definido en esos diccionarios (Sidorov, 2014).

[3] Además, se ha de indicar que las fórmulas, tal y como aparecen, no están expresadas correctamente en esta guía, por lo que ha sido necesario expresarla en términos correctos. La expresión utilizada se puede ver en la aplicación práctica

3.5 El problema de las emociones concretas: Vergüenza

Al estudiar las herramientas de análisis de sentimientos, nos damos cuenta del papel secundario que, paradójicamente, tienen las propias emociones en estas. Esta investigación inicialmente se planteó como una exploración de la identificación y el tratamiento de emociones concretas, hasta darnos cuenta de que no es posible llevarlo a cabo por la falta de estas mismas herramientas.

En este sentido, como parte de la fundamentación teórica, se tiene en cuenta la emoción de la vergüenza [4], ejemplificada con el fin de mostrar tanto la falta de la presencia de las emociones, así como para aportar una perspectiva en la que el análisis de texto y el análisis bibliométrico está de la mano no solamente del estudio de las realidades sociales, sino también de la producción científica que las trata.

Así, en primer lugar se planteó ver qué producción científica existe sobre el tema en la base de datos *Scopus* [5] (Faen Scopus, 2005) para ver la importancia concedida a este tema por la comunidad científica en general. Este estudio se llevó a cabo tras la revisión de los paquetes de análisis de sentimientos de `R` y para su estudio fue utilizado el paquete `bibliometrix`.

En este sentido, existen ciertos inconvenientes que se perciben no solamente en las técnicas de análisis de sentimientos o de texto, sino que persisten a la esfera y a la producción de herramientas metodológicas. Dentro de estas problemáticas destacamos una principal: la falta de documentación detallada sobre las herramientas.

En cuanto a la falta de documentación, hablamos no solamente de guías prácticas de su uso, sino también de transparencia de los procedimientos. La consecuencia mientras persiste este problema se reduce a que no somos capaces de *saber* qué son los resultados obtenidos.

3.5.1 Exploración bibliométrica

Los estudios sobre la vergüenza han estado presentes con anterioridad en estudios de carácter psicológico y antropológico, en cambio, la sociología ha permanecido alejada

[4]Este estudio se establece como una continuación de la investigación sobre la emoción de la vergüenza, denominada *Vergüenza como institución social* (Shevchenko, 2018). Para el presente estudio no resulta relevante cómo definamos las emociones concretas, sino cuál es su tratamiento en en el conjunto de técnicas conocidas como *análisis de sentimientos*. Por este motivo, la definición concreta de la vergüenza, así como de otras emociones, no será expuesta en el presente documento.

[5]La base de datos *Scopus* cada vez adquiere más relevancia, sobre todo en el mundo de habla hispana, por lo que ha sido escogida para el presente estudio como referencia.

de las emociones durante más tiempo. Actualmente el interés en las emociones, y más concretamente sobre las emociones secundarias o sociales, crece de forma significativa.

Para ver la naturaleza de este crecimiento, se ha desarrollado un estudio bibliométrico de la producción científica en la base de datos SCOPUS desde su creación hasta diciembre de 2018. Acotando la búsqueda y llevando a cabo un análisis exhaustivo del área, se han filtrado los resultados, seleccionando finalmente solo aquellas obras que pertenecen catalogadas como pertenecientes a las 'ciencias sociales'. Esta acotación se habría constituido por sociología exclusivamente, sin embargo, la base de datos no lo permite.

De este modo, la búsqueda llevada a cabo se ha formulado con la siguiente sintaxis:

TITLE-ABS-KEY (shame) AND (LIMIT-TO (SUBJAREA , 'SOCI'))

Se han extraído todos aquellos manuscritos que contenían la palabra 'vergüenza', combinada con los criterios de estar incluidos en 'ciencias sociales' para todas las colecciones, por lo que se trata de un universo de datos, en vez de una muestra. En este sentido, es importante tener esto en cuenta ya que no es preciso llevar a cabo ningún tipo de inferencia estadística, puesto que los resultados observados son los de la población del periodo estudiado.

Así pues, disponemos de un total de 3757 documentos escritos por un total de 6173 autores, lo que deja intuir que la correspondencia por autor será de más de un documento, es decir, habrá un conjunto de autores principales que se encargan de escribir sobre esta temática. Esto se corrobora cuando se ve que hay 1845 autores que han publicado documentos por sí solos, al mismo tiempo que hay 2143 artículos firmados por un solo autor [6].

[6]Los apartados expresados dentro de la tabla se encuentran en inglés por definición y no se encuentra disponible ningún procedimiento sencillo de modificar esta salida, por lo que no se ha traducido

Description	Results
Documents	3757
Sources (Journals, Books, etc.)	1733
Keywords Plus (ID)	3951
Author's Keywords (DE)	6461
Period	1852 - 2019
Average citations per documents	16.62
Authors	6173
Author Appearances	7059
Authors of single-authored documents	1958
Authors of multi-authored documents	4215
Single-authored documents	2143
Documents per Author	0.609
Authors per Document	1.64
Co-Authors per Documents	1.88
Collaboration Index	2.61
Document types	
ARTICLE	2806
ARTICLE IN PRESS	104
BOOK	112
BOOK CHAPTER	279
CONFERENCE PAPER	32
EDITORIAL	22
ERRATUM	5
LETTER	22
NOTE	33
REVIEW	334
SHORT SURVEY	8

A lo largo del periodo estudiado, entre 1852 y 2019, la producción científica aumenta. Se puede ver que hasta mediados de los años 80 apenas había un solo artículo por año. Sin embargo, a partir de los años 90 la producción científica sobre la emoción de la vergüenza en sociología aumenta cada vez más, llegando a tener más de 300 artículos en el año 2018.

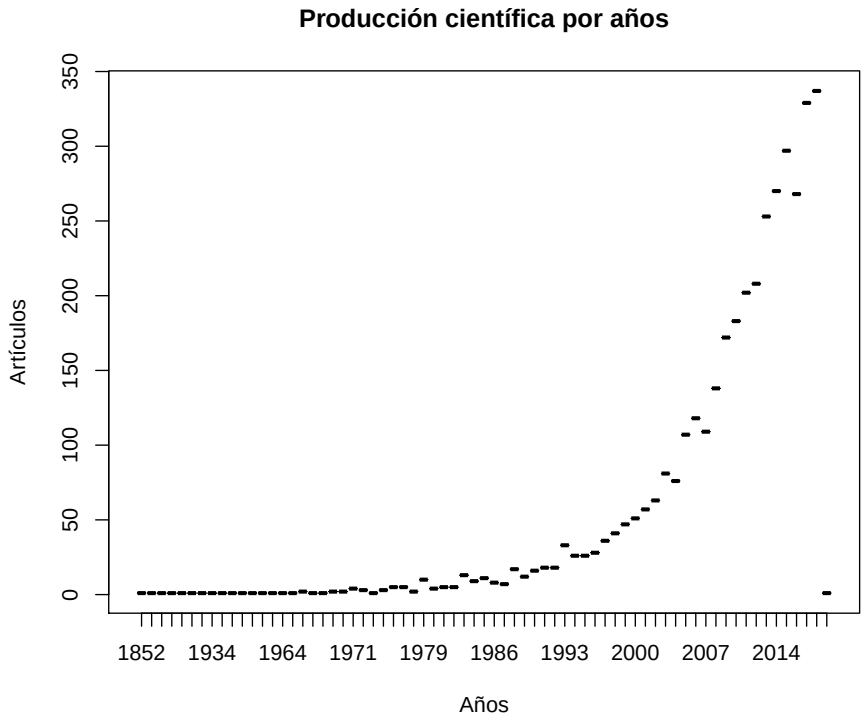

Figure 3.1: Aumento de la producción científica temática

Si consideramos las primeras 15 potencias mundiales dentro del estudio de la vergüenza por su producción, nos encontramos con que España queda fuera, estando la mayoría de estas potencias por debajo de los 200 artículos, con la excepción de UK y, sobre todo, EEUU con más de 1000 artículos.

Para conocer la realidad de los estudios de la vergüenza, se han considerado las palabras incluidas en los títulos como unidades elementales que pueden describir las temáticas con las que los autores la han relacionado. En este sentido, se han extraído las 30 palabras principales que se presentan en el conjunto de los títulos de los artículos, y se han eliminado las palabras repetidas para que cada una apareciese una sola vez [7].

Siguiendo este enfoque, se podría plantear estudiar las relaciones en función de algún indicador estadístico, en vez de tratar únicamente con las frecuencias de coincidencia. De este modo, a continuación, exponemos un gráfico en el que se muestran las relaciones entre los términos por fuerza de asociación y divididos en conglomerados, sin olvidar que todas

[7] Nota: aquellas palabras que representan lo mismo, pero que están escritas de diferentes formas aparecen repetidas, puesto que la técnica no las reconoce como semejantes. En el que caso de querer profundizar en este aspecto, sería necesario llevar a cabo una depuración de las palabras obtenidas.

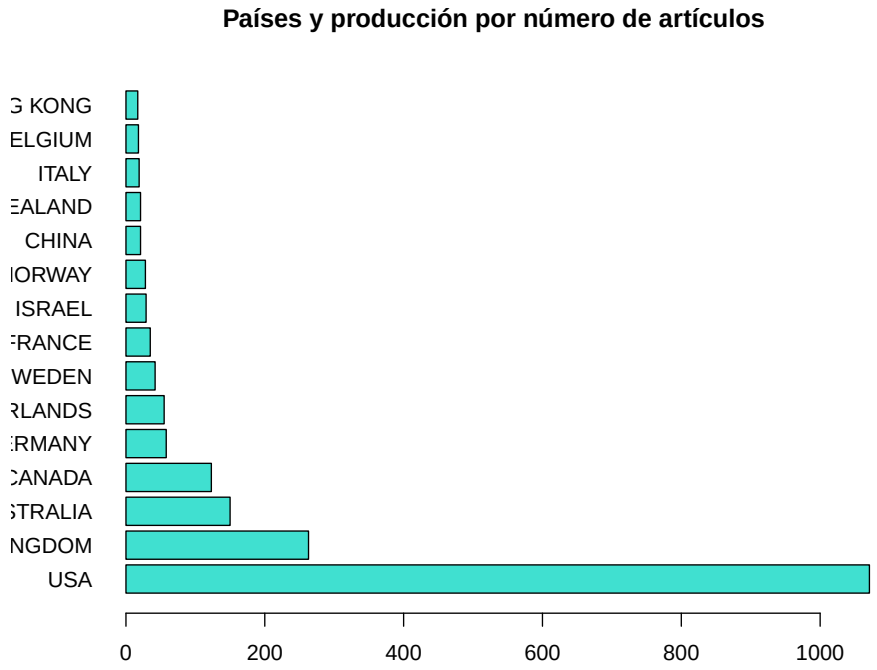

Figure 3.2: Principales productores

están relacionadas entre sí.

Se observa claramente la diferenciación en 4 conglomerados, aunque dos de ellos se establecen por la relación entre dos términos: el primero de ellos, se da entre ética y afecto y, el segundo, entre objetivación del self y la imagen del cuerpo. Los conglomerados restantes tienen conjuntos de términos destacablemente numerosos en comparación. Entre ellos encontramos el conglomerado rojo, formado por el miedo, trauma, vuilencia, culpa, ansiedad, identidad, orgullo, vergüenza, cultura y emociones morales, entre otros, y el conglomerado azul reune a la familia, el estigma, la pobreza, la violencia doméstica, la sexualidad, a la mujer y la salud mental.

Resulta de gran interés que estos 30 términos principales asociados entre sí se agrupan de una forma que resulta claramente interpretable desde la teoría sociológica, así como desde la relación con otras emociones y asociaciones a colectivos concretos. Asimismo, en su mayoría se trata de una terminología inclinada hacia la negatividad de la realidad social, todo aquello que se trata de ocultar.

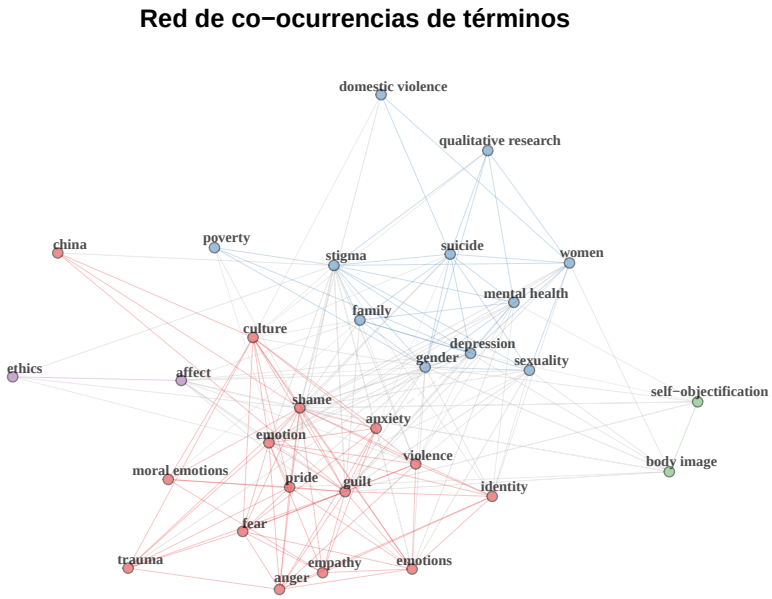

Figure 3.3: Relaciones entre los términos

3.5.2 Utilidades para el estudio de las emociones

Desde la sociología sociología de las emociones, tanto como desde la sociología en general, resulta de gran relevancia conocer a la comunidad científica que produce el conocimiento, como las temáticas que se tratan. Las metodologías de los estudios documentales para la preparación de teoría o marcos teóricos para posteriores aplicaciones o creaciones de herramientas pueden resultar demasiado costosas para los recursos de tiempo y de fuerza intelectual disponible para la resolución de problemas o desarrollo de proyectos.

En este apartado, hablando de la emoción de la vergüenza como un caso de emoción no considerada en la mayoría de las técnicas de análisis de sentimientos [8], se puede estudiar con la minería de texto incluso en un sentido temático o bien con la ayuda de técnicas bibliométricas con el fin de ser capaces de reunir una teoría fragmentada como la que existe.

[8] Por ahora, la única forma de tratamiento de la vergüenza ha sido identificada en algunos diccionarios desarrollados para inglés, que incluyen algunos términos a partir de los que se puede decir que se está expresando vergüenza. Pero no hay ninguna fundamentación teórica, ni estadística a la hora de entender qué representa esa vergüenza: quien lo dice, ¿siente vergüenza?, ¿avergüenza a otro? ,¿rechaza una norma social?, etc. ¿Es timidez o humillación?, ¿de qué grado se trata?

En este sentido, una forma de ser capaces de *estudiar* las emociones y ser capaces de *aportar* mejoras al análisis de sentimientos o a la ciencia en general, se encuentra también en el propio uso de herramientas como la minería de texto, tanto por su cuenta como incluida dentro de análisis bibliométricos.

Capítulo 4

MANIPULACIÓN DE DATOS DE TEXTO

A la hora de aproximarse al mundo del análisis de datos, se hace notorio que una de las cuestiones más complejas a priori es el manejo de los datos. Cuando hablamos del manejo o la manipulación de los datos nos referimos a cuestiones que podrían parecer sencillas, nada más lejos de la realidad. Entre estas cuestiones se encuentra:

- Ser capaces de abrir y leer los archivos desde, en este caso, `R`.
- Ser capaces de editar los datos.
- Ser capaces de organizar los datos como nos resulte más cómodo para visualizar o procesar.
- Ser capaces de transformar los datos y guardarlos para análisis posteriores.

En este capítulo, se explicará el proceso seguido para lograr la manipulación de datos del CorpusMuchoCine, pasando por la importación de los datos, la extracción de la información deseada, la organización de esta información, su preprocesamiento y, por último, el guardado de los datos preparados para ser utilizados en análisis.

Inicialmente, se cargan los paquetes que se refieren a la depuración de los textos (`tm`), la lectura de archivos '.xml' (`XML` y `xml2`) y la organización y visualización amigable de los datos (`dplyr`) [1]. A lo largo de los capítulos se cargarán los paquetes que sean necesarios en cada momento, por lo que la lista de todos ellos se podrá ver únicamente en Anexos.

[1] Con el fin de no entorpecer la lectura, las referencias de los paquetes utilizados se expondrán en la Bibliografía

```
library(tm)
library(XML)
library(xml2)
library(dplyr)
```

4.1 Obtención de los datos

No hay muchos conjuntos de datos para pruebas de análisis de texto preparados para castellano, por lo que, de no encontrarse, sería necesario recurrir a técnicas de extracción, recopilación y organización de los datos a partir de web scraping, es decir, descarga automática de texto a partir de fuentes públicas. En este caso, para evitar esa dificultad añadida, se ha procedido a contemplar qué bases de datos se han llegado a utilizar en artículos científicos especializados para castellano.

A partir de la revisión de la literatura, se ha localizado una base de datos denominada 'CorpusMuchoCine', que contiene 4380 archivos de formato '.xml', extraídos a partir de críticas de cine en la página web 'muchocine.net'. Cada uno de los archivos incluye la siguiente información:

- Nombre del usuario que ha dejado el comentario.
- El título de la película que ha comentado.
- Un resumen de su comentario (`sumario`).
- El comentario, formulado como un texto significativamente más largo (`texto`).
- La puntuación asignada a la película, de 1 (muy mala) a 5 (muy buena).

Esta base de datos se ha descargado de la página web de la Universidad de Sevilla, [2], concretamente desde el enlace a su creador, Fermín Cruz Mata. Este profesor creó el conjunto para su artículo 'Clasificación de documentos basada en la opinión: experimentos con un corpus de críticas de cine español' en 2008.

[2] http://www.lsi.us.es/~fermin/index.php/Datasets

4.2 Preparación del data frame

Para poder trabajar con estos datos, inicialmente se ha de lidiar con una serie de problemas inherentes a ellos. El primer problema a solucionar es el tratamiento de los datos que están en formato '.xml' y la forma en la que divide las diferentes variables. La decisión tomada consistió en construir un data frame con 4 variables, excluyendo el nombre de los usuarios, y guardarlo en un único archivo de tipo RData.

Encontrar una solución eficiente no es una cuestión sencilla en cuanto a que el procesamiento de estos datos con los paquetes especializados para el tratamiento con el formato '.xml' no han sido capaces de leer el texto. Esta dificultad añadida ha surgido a partir de problemas de codificación y, también, de la propia separación de los datos de texto, pues a veces el formato de '.xml' no sigue el estándar, mezclándose con el 'html'.

A la hora de tratar de importar los datos, se producían errores que no permitían la lectura de los ficheros, debido a que había caracteres que no estaban codificados en 'UTF-8'. Estos problemas se encuentran en los datos originales, siendo propios de la descarga y formulación de la base de datos con un procedimiento de web scraping.

Inicialmente, se intentó llevar a cabo una depuración con la ayuda del programa NotePad++. En cambio, esta forma de proceder era considerablemente manual y daba un resultado de aproximadamente 800 archivos legibles de un total de 4380.

Finalmente, con la ayuda de la función `read_html` del paquete `xml2`, ha sido posible leer 3878 textos y, posteriormente, convertirlos en una base de datos contenida en un único archivo. El tamaño de la base de datos ha sido reducido debido a que se ha indicado que aquellos archivos que no cumplen la condición exigida o, dicho de otro modo, den error se conformen como `NAs`, es decir, que se excluyan.

A continuación, se expresa el procedimiento concreto que se ha seguido para la conformación de esta base de datos unificada. Así, en primer lugar, se establece un directorio desde el que se importarán los archivos de trabajo y en el que se guardarán todos los arcvivos creados.

```
basePath <- './corpusCriticasCine' # Directorio de trabajo


# Patrón de archivos que la función tendrá que procesar
files <- dir(basePath, pattern = '*.xml')
```

Se crea una base de datos vacía con un número de filas de tanta longitud como tenga el número de archivos que se ha tomado a partir del objeto 'files' y con cuatro columnas, cuyos nombres se indican. Este procedimiento permite que los datos sean organizados de una forma computacionalmente eficiente,además de tener una forma específica decidida en relación a los tratamientos posteriores que se llevarán a cabo.

```
# Creamos una dataframe vacío para guardar los datos
n <- length(files);
data <- data.frame(matrix(NA, nrow = n, ncol = 4))
names(data) <- c('puntuacion', 'titulo', 'sumario', 'texto')
```

El paso siguiente consiste en crear una función que abra y lea cada uno de los archivos originales en '.xml' en la carpeta donde se encuentran y extraiga la información dentro de las variables, sin tener en cuenta la estructura de estos archivos.

Esto se consigue indicándole que, para el caso de la variable `sumario` extraiga aquello que en '.xml' está contenido en el apartado de 'xpath' como `res$summary`, teniendo en cuenta que el original tiene una codificación 'Windows-1252'.

Esta función, por otro lado, indica en cada variable qué naturaleza ha de tener la nueva variable que se ha de conformar, es decir, si va a ser de tipo 'caracter' (texto) o de tipo numérico. También se indica que este procedimiento se haga para cada archivo, siguiendo un objeto llamado 'contador', es decir, a partir del primer archivo, la siguiente acción se hará sobre el número de ese archivo +1, lo que viene a indicar el archivo inmediatamente posterior.

```
# Organizamos y escribimos la información dentro del data frame creado
contador <- 1


for (f in files) {
 tmp <- read_html(file.path(basePath, f), encoding = 'Windows-1252')
 res <- as_list(tmp)$html$body$review
 data$sumario[contador] <- as.character(res$summary)
 data$texto[contador] <- if (length(res)>1) as.character(res[[2]])
 else NA_character_
 data$titulo[contador] <- attributes(res)$title
 data$puntuacion[contador] <- as.numeric(attributes(res)$rank)
```

```
 contador <- contador + 1}
```

Por último, indicamos que se guarde la información creada en las cuatro variables en el directorio de trabajo con el nombre 'data' y formato propio de R 'RData'.

```
# Guardamos la base de datos en el directorio
save(data, file = '../data.RData')
```

Capítulo 5

PREPROCESAMIENTO

Una vez que los datos están creados y guardados, se procede a su importación a la sesión de R. Esto se lleva a cabo para no tener que procesar cada vez la información desde 0, sino evitar este proceso pudiendo obtener los datos preparados de forma cómoda y computacionalmente sencilla.

```
load('../data.RData')
```

La función `str` permite ver un resumen tanto del contenido de cada variable como de su naturaleza y longitud. En este caso, vemos que se trata de un data frame con 4 variables, una numérica y 3 de texto, y 3878 observaciones para cada una de ellas. Además, se puede ver cómo los datos siguen presentando ciertos problemas, como el caso de la variable `texto`, que sigue conteniendo símbolos indeseados. En cambio, en este punto no le prestamos atención, teniendo en cuenta que en el procesamiento posterior estos detalles serán depurados.

```
# Vemos las características de las variables obtenidas
str(data, width = 70, strict.width = 'wrap')
```

```
## 'data.frame':    3878 obs. of  4 variables:
## $ puntuacion: num 4 4 1 2 4 1 3 4 4 3 ...
## $ titulo : chr "May" "Paradise now" "La alianza del mal" "Pequeños
##    grandes héroes" ...
## $ sumario : chr "May, ¿quieres ser mi amigo?" "Cómo ponerse en la
##    piel de un kamikaze" "Silicona, esteroides, pactos demoníacos y
##    otras basuras habituales son la base que sustentan esta
```

```
##    aberración. De vergüenza." "Una comedia entretenida y poca cosa
##    más para ver una tarde de domingo" ...
## $ texto : chr "\"May, ¿Quieres ser mi amigo?\" es una de esas
##    películas que nos recuerdan que el terror no siempre lleva garra"|
##    __truncated__ "Es todo un alivio que ante tanta película que trata
##    el tema palestino de una forma un tanto lineal, posicionada"|
##    __truncated__ "Una fiesta llena de excesos, rubias despampanantes,
##    musculitos por doquier, algún que otro muerto. nada nuevo. "|
##    __truncated__ "Zoom nos cuenta la historia de Jack Shepard,
##    anteriormente conocido como el Capitán Zoom, Superhéroe que
##    perdió"| __truncated__ ...
```

La librería `tm` es un conjunto de funciones centradas en el procesamiento de texto como una forma de iniciar minería de texto. Dentro de estas funciones se encuentran la creación de un tipo de objeto, denominado `corpus` para depurar y organizar la información, así como crear matrices de términos a partir de los datos originales (Feinerer, 2018).

5.1 Preprocesamiento de la variable 'sumario'

Disponemos de dos variables de descripción de la opinión: sumario y texto. Habitualmente en los estudios de análisis de sentimientos u 'opinion mining' se procesan datos de redes sociales como Twitter, que tienen límite de letras para los comentarios.

Sin embargo, este enfoque, aunque computacionalmente más eficiente, puede complicar la determinación de las emociones o de la dirección de una opinión debido precisamente a que la cantidad de las palabras utilizadas es pequeña. En este sentido, este trabajo tiene como finalidad, entre otras, observar las diferencias de la capacidad de detección de aspectos emocionales en textos de diferente tamaño. En el caso de la variable `texto` se trata de párrafos, en principio sin límite de extensión.

De este modo, la primera parte del preprocesamiento del dataset se centra específicamente en la depuración de la variable `sumario`. En primer lugar, se crea un objeto que hará referencia a la variable concreta, lo que nos permitirá referenciarla sin tener que recurrir a través de la base de datos, es decir, únicamente poniendo su nombre.

```
sumario2 <- data$sumario; length(sumario2)
```

```
## [1] 3878
```

Se crea un objeto de tipo `Volatile corpus` para la variable `sumario`. Se trata de un tipo de transformación de datos, ya que 'corpus' tiene en cuenta, además del texto inicialmente introducido, una serie de variables a mayores. Por otro lado, se caracteriza, como su nombre indica, por la 'volatilidad', es decir, los datos contenidos y creados en el 'corpus' se borran al borrarse el objeto al que estaban asignados .

```
# Creación de un objeto 'corpus' para aplicar 'tm'
corpus_sumario <- VCorpus(VectorSource(sumario2))
# Comprobación del funcionamiento correcto del corpus
content(corpus_sumario[[5]]) # Contenido concreto del elemento 5
```

```
## [1] "Luc Besson sabe manejar la acción, y aquí lo demuestra de nuevo
 manteniendo todo el film un ritmo trepidante sin apenas descanso."
```

Para que sea posible analizar el texto de forma conjunta, es necesario estandarizar los datos.. En este sentido, las principales diferencias que nos podemos encontrar provienen de una serie de diferenciaciones, imprescindible para entendernos en el lenguaje escrito y hablado, pero que son ruido a la hora de llevar a cabo el análisis.

Así pues, la función `tm_map` del paquete `tm` nos permite transformar los datos convirtiendo todo el texto en escritura a minúsculas, sin tildes, sin espacios en blanco y, por último, sin símbolos de puntuación. Esto último, a su vez, resuelve posibles problemas de compatibilidad de encodings que en futuras etapas.

```
    # Estandarización del texto: Transformaciones

# Conversión a minúsculas
corpus_sumario2 <- tm_map(corpus_sumario, content_transformer(tolower))
# Eliminar símbolos de puntuación
corpus_sumario2 <- tm_map(corpus_sumario2, removePunctuation)
# Eliminar las puntuaciones no incluidas por defecto
corpus_sumario2 <- tm_map(corpus_sumario2, content_transformer(gsub),
                          pattern = '[¿¡]+', replacement='')
```

```
# Eliminar números
corpus_sumario2 <- tm_map(corpus_sumario2, removeNumbers)
# Eliminar tildes
corpus_sumario2 <- tm_map(corpus_sumario2, content_transformer(chartr),
                          old = 'áàéèíìóòúù',new ='aaeeiioouu')
# Eliminar espacios en blanco
corpus_sumario2 <- tm_map(corpus_sumario2, stripWhitespace)
```

En el apartado anterior mencionábamos la existencia de una serie de 'ruido' que se encuentra en los datos y dificulta el análisis, sin aportar información. Este concepto no se aplica únicamente al estilo de escritura, sino a una serie de palabras que por sí solas no tienen un significado específico. En lenguaje de minería de texto estas palabras se denominan `stopwords`.

Existen diccionarios específicos de stopwords por lenguaje, por ejemplo, los artículos en castellano, pero también según la temática que se está tratando. En este caso concreto, eliminaremos únicamente las que se refieren al idioma, puesto que el objetivo que se persigue es observar las técnicas y no extraer conclusiones significativas sobre los datos.

Por otro lado, no resulta preciso aplicar la eliminación de las stopwords sin tener en cuenta que estas, en principio, no han sido estandarizadas (pueden llevar tildes, por ejemplo). Por lo que, además de añadir unas muy pocas que aparecían muy frecuentemente y no se encontraban en la lista, como 'aunque', se procesan.

Es importante volver a eliminar los espacios en blanco una vez que se extraen las stopwords, ya que estos pueden producir problemas en la lectura del texto resultante.

```
stopwords('spanish')[1:10] # 10 stopwords en castellano
```

```
##  [1] "de"  "la"  "que" "el"  "en"  "y"   "a"   "los" "del" "se"
```

```
# Normalización de stopwords: eliminación de tildes
stopwords2 <- chartr(x = stopwords('spanish'), old = 'áàéèíìóòúù',
                     new = 'aaeeiioouu')
# Eliminación de stopwords
corpus_sumario3 <- tm_map(corpus_sumario2, removeWords, stopwords2)
# Creación de una lista de stopwords adicionales
```

```
add_stopwords <- c('aunque', 'ver', 'puede', 'ser')
# Eliminación de stopwords adicionales
corpus_sumario3 <- tm_map(corpus_sumario3,removeWords,add_stopwords)
# Eliminar espacios en blanco surgidos por la eliminación
corpus_sumario3 <- tm_map(corpus_sumario3, stripWhitespace)
```

Además del preprocesamiento llevado a cabo, existen otras herramientas que muchas veces se aplican a los datos de texto, como por ejemplo, *stemming*. Este procedimiento consiste en reducir las palabras presentes a sus raíces, lo que se suele resumir en eliminar la terminación. En este caso, no es recomendable aplicarlo porque los diccionarios están conformados con palabras enteras, tanto los que se utilizan para detectar el contenido emocional, como aquellos que aportan las stopwords o semejantes.

Finalmente, podemos ver un ejemplo de las diferencias que se presentan entre el texto antes y después de procesar. Para ello, vemos el primer sumario del comentario que se encuentra en el dataset:

```
data$sumario[1]
```

```
## [1] "May, ¿quieres ser mi amigo?"
```

```
content(corpus_sumario3[[1]])
```

```
## [1] "may quieres amigo"
```

5.2 Preprocesamiento de la variable `texto`

A continuación, se lleva a cabo el mismo procedimiento que para la variable sumario, por lo que no requiere explicación del código aplicado. La diferencia principal reside en la longitud del texto y, por otro lado, por la capacidad computacional que requiere su manipulación. Esto es, a mayor longitud del texto, así como complejidad, el procesamiento lleva más tiempo.

```
texto2 <- data$texto; length(texto2) # Eliminar casos vacíos
```

```
## [1] 3878
```

```r
# Creación de un objeto 'corpus' para aplicar 'tm'
corpus_texto<- VCorpus(VectorSource(texto2))
# Comprobación del funcionamiento correcto del corpus
content(corpus_texto[[5]]) # Contenido concreto del elemento 5
```

```
## [1] "Luc Besson dirige esta película basada en sus propios libros ..."
```

```r
# Estandarización del texto: Transformaciones

# Conversión a minúsculas
corpus_texto2 <- tm_map(corpus_texto, content_transformer(tolower))
# Eliminar símbolos de puntuación
corpus_texto2 <- tm_map(corpus_texto2, removePunctuation)
# Eliminar las puntuaciones no incluidas por defecto
corpus_texto2 <- tm_map(corpus_texto2, content_transformer(gsub),
                        pattern = '[¿¡]+', replacement='')
corpus_texto2 <- tm_map(corpus_texto2, removeNumbers) # Números
# Eliminar tildes
corpus_texto2 <- tm_map(corpus_texto2, content_transformer(chartr),
                        old = 'áàéèíìóòúù',new ='aaeeiioouu')
# Eliminación de stopwords
corpus_texto3 <- tm_map(corpus_texto2, removeWords, stopwords2)
# Eliminar stopwords añadidas
corpus_texto3 <- tm_map(corpus_texto3,removeWords,add_stopwords)
# Eliminar espacios en blanco
corpus_texto3 <- tm_map(corpus_texto3, stripWhitespace)
```

5.3 Combinación y guardado de los datos procesados

Una vez que los datos están listos para ser analizados, la naturaleza del 'VCorpus' nos obliga a crear un archivo que los contenga, ya que de no ser así, al cerrar sesión se perdería el avance llevado a cabo. Esto solo es necesario para el análisis de sentimientos, ya que para otros análisis las herramientas no están implementadas en el paquete `tm`.

Este proceso tiene como dificultad principal la naturaleza del objeto que se produce como

resultado de la conversión y del procesamiento del `corpus`. Esto es, el resultado es una lista de elementos, dentro de los cuales solamente necesitaremos un elemento: `content`.

Si aplicamos la función `str()` al primer elemento del objeto `corpus_sumario3`, obtenemos el resultado que nos indica la cantidad de información que se ha recopilado y asociado.

```
str(corpus_sumario3[1], width = 80, strict.width = 'wrap')
```

```
## List of 1
## $ 1:List of 2
## ..$ content: chr "may quieres amigo"
## ..$ meta :List of 7
## .. ..$ author : chr(0)
## .. ..$ datetimestamp: POSIXlt[1:1], format: "2019-09-06 09:17:54"
## .. ..$ description : chr(0)
## .. ..$ heading : chr(0)
## .. ..$ id : chr "1"
## .. ..$ language : chr "en"
## .. ..$ origin : chr(0)
## .. ..- attr(*, "class")= chr "TextDocumentMeta"
## ..- attr(*, "class")= chr [1:2] "PlainTextDocument" "TextDocument"
## - attr(*, "class")= chr [1:2] "VCorpus" "Corpus"
```

El problema consiste en que no solamente se ha de tratar con una lista asociada al resultado, sino que esta lista está creada para cada elemento por separado. Por este motivo, es preciso formular una función que extraiga el contenido del elemento 'content' de cada uno de los comentarios del corpus.

Para ello, se crean dos objetos vacíos, que se ocuparán con los resultados de la extracción de 'content' a partir de cada elemento del corpus, tanto para sumario como para texto.

```
# Extracción del contenido procesado para 'sumario'
new_sumario <- c() # Creamos objeto vacío para guardar los datos
for (i in 1:length(corpus_sumario3)){
 new_sumario[i] <- corpus_sumario3[[i]]$content
}
```

```
# Extracción del contenido procesado para 'texto'
new_texto <- c()
for (i in 1:length(corpus_texto3)){
  new_texto[i] <- corpus_texto3[[i]]$content
}
```

Por último, una vez que los objetos `new_sumario` y `new_texto` contienen las listas del `content` procesado anteriormente, es preciso combinarlos con las variables originales no modificadas en una nueva base de datos `RData`. Para ello, primeramente se crea un nuevo data frame con las cuatro variables y, finalmente, se guarda en el directorio de trabajo bajo el nombre de `df_procesado.RData`.

```
df_procesado <- data.frame(data$puntuacion, data$titulo, new_sumario,
                           new_texto, stringsAsFactors = FALSE)

save(df_procesado, file = 'df_procesado.RData')
```

Capítulo 6

CREACIÓN DE LA MATRIZ DE TÉRMINOS

Para poder llevar a cabo un análisis exploratorio de los datos, es necesario trasladarlos a una naturaleza que sea procesable con métodos estadísticos. En este sentido, las palabras como tal no pueden ser tratadas cuantitativamente, pero se puede crear una base de datos que contenga la frecuencia de su aparición en relación a los demás términos.

La función `DocumentTermMatrix` de la librería `tm` sirve para generar una variable (columnas) para cada término/palabra que aparece en los textos, que contiene la frecuencia de aparición en cada documento (filas).

Como es una función del paquete `tm`, se aplica a los corpus procesados número 3, tanto de la variable `sumario` como de `texto`. Además, si aplicamos la función `inspect` a estos objetos, podemos ver la cantidad de términos que se está considerando o, dicho de otra forma, la cantidad de variables que se han extraído.

Así, en este caso, para la variable `sumario` se han extraído en total 13808 términos, llegando a haber palabras de longitud de caracteres 23, lo cual parece sugerir alguna clase de error. Para el caso de la variable `texto`, la cantidad de términos ha sido de 81756 y la longitud de término máxima 53 caracteres.

```
dtm_sumario <- DocumentTermMatrix(corpus_sumario3)
dtm_texto <- DocumentTermMatrix(corpus_texto3) # Dtm de 'texto' procesado
```

6.1 Reducción de dimensiones

Al tratar con cada palabra en cada comentario, a pesar de que los datos estén depurados, se crea un número alto de variables. Estas variables, a su vez, en su inmensa mayoría contendrán el valor '0' como resultado. Esto es, la mayor parte de los términos no aparecerán en la mayor parte de los comentarios.

Por lo tanto, el interés a la hora de conformar estos datos reside en reducir al máximo posible, manteniendo la mayor parte de la información, el volumen de datos y evitar posibles errores, como pueden ser palabras de más de 15 o 20 caracteres o semajantes.

Para paliar con este volumen de datos, vamos a tratar principalmente con dos procedimientos:

- El establecimiento de la longitud mínima y máxima de las palabras
- Eliminación de las palabras menos frecuentes

En primer lugar, se establece como límite inferior la cantidad de 2 caracteres y como límite superior 15 para las dos variables a partir de las cuales se han creado las matrices de términos. Al guardarse el 'dtm_sumario2' como nuevo objeto, las palabras que no cumplen con esta condición no se incluyen dentro de su matriz.

Se puede ver que hubo cierta reducción, siendo esta mayor en la variable `texto` (79077 de 81756), ya que era la que tenía palabras de marcada longitud, frente a la variable sumario (13774 de 13808). La reducción en ambos casos ha sido moderada, lo que indica que los términos excesivamente cortos o largos se daban en muy pocos casos.

```
dtm_sumario2 <- DocumentTermMatrix(corpus_sumario3,
                                   list(wordLengths = c(2,15)))


dtm_texto2 <- DocumentTermMatrix(corpus_texto3,
                                 list(wordLengths = c(2,15)))
```

El segundo procedimiento está enfocado hacia lo que se denomina `sparsity` o la dispersión de los términos del documento. Esta dispersión va de 0, ninguna dispersión, a 1, toda la dispersión posible. Esto significa que la matriz creada original tiene una `sparsity` de 1, ya que conserva todos los términos originales como variables, sean estos o no mínimamente frecuentes.

Para reducir ligeramente el volumen de datos, puesto que una limitación en este aspecto resulta notoria, establecemos un 0,99 o 99% de dispersión como valor a partir del cual las palabras muy poco frecuentes en menos de 1% de los documentos sean eliminadas de la matriz de términos. Esto es, se permite como máximo una `sparsity` de 0,99.

Las nuevas matrices de términos, habiendo eliminado las palabras menos frecuentes, se han reducido significativamente, lo que se puede observar a partir de los resultados de `inspect`: han permanecido 186 términos para la variable `sumario` y 3448 para la variable `texto`.

```
dtm_sumario3 <- removeSparseTerms(dtm_sumario2, 0.99)
inspect(dtm_sumario3)
```

```
## <<DocumentTermMatrix (documents: 3878, terms: 186)>>
## Non-/sparse entries: 15475/705833
## Sparsity           : 98%
## Maximal term length: 15
## Weighting          : term frequency (tf)
## Sample             :
##       Terms
## Docs   bien cine cinta final guion historia mejor pelicula peliculas tan
##   129     0    0     0     0     0        0     0        2         0   0
##   1605    0    0     0     0     0        0     0        0         0   0
##   2638    0    1     0     0     0        0     0        0         0   0
##   2675    2    0     0     1     0        0     0        1         0   0
##   2752    0    0     0     0     0        0     0        1         0   1
##   3297    0    0     0     0     0        0     0        0         0   0
##   34      2    0     0     0     0        0     1        0         0   0
##   688     0    1     0     0     0        0     0        0         1   0
##   769     0    0     1     0     0        0     0        1         0   0
##   947     0    0     0     0     0        0     0        1         1   0
```

```
dtm_texto3 <- removeSparseTerms(dtm_texto2, 0.99)
inspect(dtm_texto3)
```

```
## <<DocumentTermMatrix (documents: 3878, terms: 3448)>>
## Non-/sparse entries: 488050/12883294
## Sparsity           : 96%
```

```
## Maximal term length: 15
## Weighting          : term frequency (tf)
## Sample             :
##       Terms
## Docs   bien cine dos hace historia pelicula personajes solo tan vez
##   1102    3    9   1    3        4        2          7    3   8   4
##   229     3    5   7    4        6       22          5    3   8  14
##   2683    4    1   1    1        4        6          0    6   4   3
##   2950    1    6   0    2        0        1          2    5   4   2
##   2999    0    3   5    1        8       19          1    2   2   2
##   3217    3    3  16    1        4        3          2    0   2   6
##   3233    1    6   7    5        3       22          7    3   4   1
##   333     1    1   2    4        6        8          0    7   5   3
##   3843    6    9   4    1        3       29         12    5   5   5
##   43      2    4   3    2        1        3          2    0   7   9
```

6.2 Conversión de la matriz DTM a data frame

A pesar del correcto funcionamiento del paquete `tm` y de sus utilidades, existen tanto problemas de compatibilidad de sus formatos con otros paquetes, así como de tiempo de computación. Por lo tanto, si se quieren emplear otras herramientas no implementadas en el paquete `tm` será necesario convertir los datos al formato estándar de `R` (`data.frame`).

En este sentido, resulta preciso en todo caso y cuanto antes, extraer los datos procesados y convertirlos a un formato fácilmente accesible desde R, sin que se tenga que llevar a cabo este preprocesamiento innumerables veces, siempre que queramos hacer pruebas o modificar aspectos determinados.

En este caso, para crear la matriz de términos era necesario recurrir a la formulación de la base de datos como `corpus` y para evitar este procedimiento, los datos se transforman en un data frame.

```
df_dtm_sum <- data.frame(as.matrix(dtm_sumario3))
df_dtm_text <- data.frame(as.matrix(dtm_texto3))
```

Finalmente, se guardan estas matrices en un archivo externo de formato 'RData' en el directorio establecido y, siempre que sea posible, se trabajará con estos datos ya conformados.

```
save(df_dtm_sum, file = 'df_dtm_sum.RData') # DTM sumario
save(df_dtm_text, file = 'df_dtm_text.RData') # DTM texto
```

Capítulo 7

ANÁLISIS DESCRIPTIVO

Una vez que los datos están preparados para la aplicación de análisis estadísticos, procedemos a su exploración y descripción de sus principales características.En este sentido, teniendo en cuenta las peculiaridades de los datos, se llevará a cabo tres procedimientos específicos:

1. Exploración de la distribución de frecuencias de la variable 'puntuación', así como su representación gráfica con 'barplot'
2. Exploración de las palabras más frecuentes de la matriz de términos y su representación gráfica con nubes de palabras 'wordcloud'
3. Exploración de las correlaciones existentes entre los términos considerados

7.1 Exploración de la variable 'puntuación'

Así, dentro de la variabilidad de datos de los que disponemos, la única variable directamente observable en su sentido original es 'puntuación'. Para analizarla, vamos a trabajar con el conjunto de datos que ha sido guardado como `df_procesado`.

```
load('../df_procesado.RData')
```

Primeramente, con la función `summary` o resumen numérico, se pueden observar las principales características de esta variable, como, en este caso, el mínimo (1), el máximo (5) y la media (3,048), entre otros. Así, se deduce que se trata de una variable que ordinal de 1 a 5, donde 1 califica a la película comentada como 'muy mala' y 5 como 'muy buena', siendo el 3 la opción neutra.

```r
summary(data$puntuacion)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   2.000   3.000   3.048   4.000   5.000
```

En primer lugar, se calculan las frecuencias absolutas y se relativizan, de forma que sea posible representar las proporciones gráficamente después.

```r
frec <- table(data$puntuacion)
perc <- round(100 * frec / sum(frec), 1); perc
```

```
##
##    1    2    3    4    5
##  9.1 23.8 32.3 22.9 11.9
```

A continuación, vemos la distribución de las frecuencias representada en un gráfico de barras. La distribución de las categorías recuerda una distribución normal, ya que la categoría mayoritaria es la central y las frecuencias van disminuyendo conforme se acercan a los valores extremos.

Por lo tanto, la mayor parte de las opiniones, 32,3%, se encuentran en la puntuación 3 o neutral, y las puntuaciones 'muy malas' representan un 9,1% frente a un 11,9% de muy buenas.

```r
barplot(
  perc,
  xlab = 'Puntuaciones',
  ylab = '%',
  col = "turquoise",
  cex.names = T,
  names.arg = c('1 (9,1%)', '2 (23,8%)','3 (32,3%)','4 (22,9%)',
                '5 (11,9%)'),
  border = F)
```

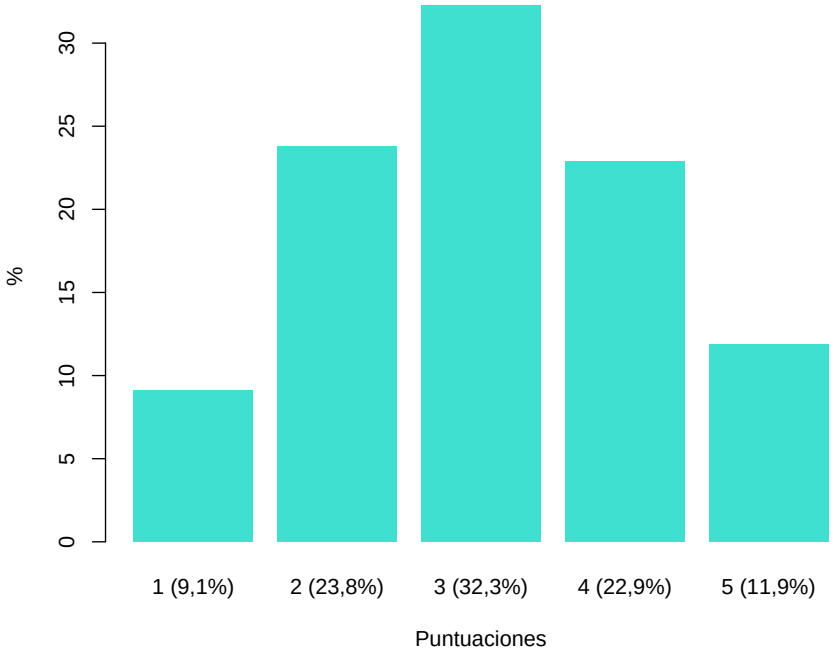

Figure 7.1: Frecuencia relativa de las diferentes puntuaciones

7.2 Exploración de las frecuencias de términos

Para comenzar la exploración de las matrices de términos, procedemos a la importación de los datos guardados a partir del procesamiento con `tm` en el capítulo anterior.

```
load('df_dtm_sum.RData') # Sumario
load('df_dtm_text.RData') # Texto
```

7.2.1 Los términos más frecuentes

A continuación, para determinar los términos más frecuentes no es necesario ya llevar a cabo cálculos, sino más bien ordenar los datos de los que disponemos a partir de las matrices. Según como estas matrices están formuladas, el sumatorio de los valores que se encuentran en cada columna es el resultado de la frecuencia absoluta de cada término.

Este procedimiento es el que se aplica con la función `sort` y se indica que el orden deseado es 'decreciente', es decir, de mayor a menor.

```
term_frecuentes_sum  <- sort(colSums(df_tm_sum), decreasing = T)
term_frecuentes_text  <- sort(colSums(df_tm_text), decreasing = T)

term_frecuentes_sum[1:15] # 15 términos más frecuentes de sumario
```

```
##  pelicula      cine  historia      bien     guion
##      1231       596       331       243       218
##  tan peliculas     mejor     final     cinta     buena      solo
##  216       207       205       186       184       176       172
##  hace      gran      buen
##   169       168       158
```

7.2.2 Las nubes de palabras

La nube de palabras es una representación gráfica que permite observar de forma muy intuitiva la terminología que más ha aparecido dentro de la matriz. Asimismo, llevarla a cabo requiere la utilización de una serie de paquetes que no se habían usado anteriormente, por lo que se activan.

```
library('SnowballC')
library('wordcloud')
library('RColorBrewer')
```

Los elementos necesarios a indicar en la formulación de la nube son los nombres o las palabras y su frecuencia. Así, para combinar estas dos variables creamos un data frame que incluya la frecuencia de términos, `term_frecuentes_sum`, por ejemplo, y establecemos esta misma como su frecuencia.

```
nube_sum <- data.frame(word = names(term_frecuentes_sum),
                       freq = term_frecuentes_sum)

nube_text <- data.frame(word = names(term_frecuentes_text),
                        freq = term_frecuentes_text)

wordcloud(nube_sum$word, nube_sum$freq,scale = c(4,.5), random.order = F,
          ordered.color = T, colors = "blue", use.r.layout = T)
```

Figure 7.2: Nubes de palabras

```
wordcloud(nube_text$word, nube_text$freq, scale = c(5,.25), max.word = 200,
          random.order = F, ordered.colors = T, colors = "red",
          use.r.layout = T)
```

7.3 Correlaciones entre términos DTM

Por último, en relación al análisis exploratorio, veremos la existencia de correlaciones entre los términos. Esta medición, en términos de r de Pearson, indica si existe relación, además de su fuerza y su dirección.

Primeramente, se crean unas matrices de correlación de todos los términos con los demás con la función `cor`. Tal y como está formulado, los resultados presentan la correlación de cada una de las palabras con todas las demás [1], por lo que no resulta posible su lectura.

[1] No se mostrarán los resultados con esta formulación, atendiendo a la ineficiencia del formato

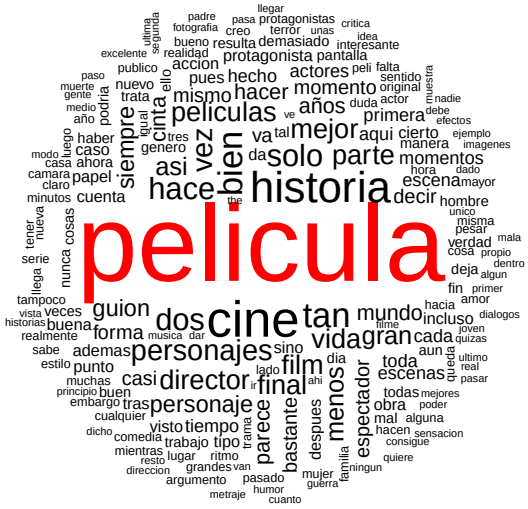

Figure 7.3: Nubes de palabras

```
correlaciones_sum <- cor(df_tm_sum, method = 'pearson')
correlaciones_text <- cor(df_tm_text, method = 'pearson')


# Con el paquete `tm`
findAssocs(dtm_sumario3, "pelicula", corlimit = 0.3)
```

```
## $pelicula
## numeric(0)
```

```
findAssocs(dtm_texto3, "pelicula", corlimit = 0.3)
```

```
## $pelicula
## numeric(0)
```

Para presentar los resultados de una forma que facilite la lectura, se ha de llevar a cabo una reorganización de los mismos. Con este fin, cargamos una serie de paquetes que incluyen herramientas para la manipulación de datos, `tidyr`, `tibbley` `dplyr`, y, por otro lado, el paquete `Hmisc`, que ofrece un cálculo de las correlaciones más detallado en la función `rcor`.

```
library(Hmisc)
library(tidyr)
library(tibble)
library(dplyr)
```

Creamos una función que distribuye las matrices de correlaciones resultantes de la función anterior en dos columnas: la primera de ellas, `row`, representa los valores que se cruzaban con una misma palabra por filas y la segunda, `column`, es la palabra que con la que se comparaban las demás.

```
# Organización del data frame resultante de 4 columnas:
# término 1, 2, coeficiente r y p valor

cor_mat <- function(cor_r, cor_p){
  cor_r <- rownames_to_column(as.data.frame(cor_r), var = 'row')
  cor_r <- gather(cor_r, column, cor, -1)
  cor_p <- rownames_to_column(as.data.frame(cor_p), var = 'row')
  cor_p <- gather(cor_p, column, p, -1)
  cor_p_matrix <- left_join(cor_r, cor_p, by = c('row', 'column'))
  cor_p_matrix
}
```

Una vez que la función organizadora está formulada, se calculan de nuevo las correalaciones, pero esta vez con la función `rcorr`, que ofrece, además de los valores de correlación un p valor como prueba de constraste de significatividad de la misma.

Solamente se debe considerar el triángulo superior de la matriz, teniendo en cuenta que es simétrica. En este sentido, siempre van a aparecer duplicados.

```
cor_sum <- rcorr(as.matrix(df_tm_sum)) # DTM de 'sumario'
mcor_sum <- cor_mat(cor_sum$r, cor_sum$P)

cor_text <- rcorr(as.matrix(df_tm_text)) # DTM de 'texto'
mcor_text <- cor_mat(cor_text$r, cor_text$P)

head(mcor_sum) # Presentación del resultado
```

```
##        row   column          cor         p
## 1 aburrida aburrida  1.000000000        NA
## 2    acaba aburrida -0.010923666 0.4964685
## 3   accion aburrida -0.007075593 0.6595851
## 4  actores aburrida  0.002683953 0.8673026
## 5   ademas aburrida  0.004338605 0.7870874
## 6  amantes aburrida  0.007433023 0.6435520
```

Una vez que los datos están organizados, podemos explorar las relaciones existentes entre los términos. Así, con la intención de ver qué términos son los que están más relacionados entre sí, de forma tanto positiva como negativa, procederemos a darle un orden lógico a las columnas.

```
# Establecimiento de un orden descendiente
mcor_sum_desc <- mcor_sum[with(mcor_sum, order(-mcor_sum$cor)), ]
mcor_text_desc <- mcor_text[with(mcor_text, order(-mcor_text$cor)), ]

# Exlusión de los casos en los que r = 1
mcor_sum_desc <-  mcor_sum_desc %>% filter(cor != 1)
mcor_text_desc <-  mcor_text_desc %>% filter(cor != 1)
```

Siguiendo este procedimiento, se pueden ver las relaciones más fuertes, tanto de forma positiva como negativa,aunque en cuanto a la significatividad de los valores p, se ha de poner en duda, teniendo en cuenta la cantidad de variables consideradas.

```
# Las 15 correlaciones más fuertes de 'sumario'
mcor_sum_desc[1:10,]
```

```
##          row     column       cor p
## 1  especiales    efectos 0.8421792 0
## 2    efectos especiales 0.8421792 0
## 3       obra    maestra 0.7576595 0
## 4    maestra       obra 0.7576595 0
## 5       rato      pasar 0.5891669 0
## 6      pasar       rato 0.5891669 0
## 7       pena     merece 0.5583780 0
## 8     merece       pena 0.5583780 0
```

```
## 9       horas        dos 0.4561500 0
## 10        dos      horas 0.4561500 0
```

```
# Las 15 correlaciones más fuertes de 'texto'
mcor_text_desc[1:10,]
```

```
##        row  column       cor p
## 1  portman natalie 0.8799132 0
## 2  natalie portman 0.8799132 0
## 3   sonora   banda 0.8753547 0
## 4    banda  sonora 0.8753547 0
## 5     wars    star 0.8701857 0
## 6     star    wars 0.8701857 0
## 7    woody   allen 0.8518957 0
## 8    allen   woody 0.8518957 0
## 9  nicolas    cage 0.8261563 0
## 10    cage nicolas 0.8261563 0
```

```
cor(df_tm_sum[,'pelicula'], df_tm_sum[,'mala'])
```

```
## [1] 0.08469343
```

Capítulo 8

ANÁLISIS DE SENTIMIENTOS CON `POLARITY`

El único procedimiento encontrado para la determinación de una puntuación *emocional* a partir de una técnica de análisis de sentimientos se encuentra en la función `polarity` del paquete `qdap`. Esta función se incluye en la parte de las técnicas que tratan con datos no supervisados desde una perspectiva de análisis semántico.

Este estimador de la `polaridad` emocional presentada en un determinado texto supone una ventaja más allá de la disponibilidad y transparencia de las fórmulas, y es que tiene en cuenta no solamente las palabras positivas y negativas para el recuento y el cálculo de la puntuación, sino también considera que la carga de las palabras se puede *invertir*, *disminuir* o *ampliar*.

La atención hacia los *modificadores* de la carga presente en el lenguaje comienza a principios de los años 2000, cuando los investigadores se dan cuenta de que la presencia de palabras negativas cerca de las palabras positivas modifican la carga positiva (Das y Chen, 2001; Pang, Lee y Vaithyanathan, 2002). En este sentido,se han tomado dos direcciones diferenciadas a la hora de tratar con esta cuestión:

- Por un lado, se desarrolla una perspectiva que da una puntuación determinada a cada palabra que implique emoción positiva o negativa, habitualmente puntuadas como +1 y -1 respectivamente [1], y, cuando una palabra positiva aparece seguida de una

[1] Las puntuaciones pueden variar según la palabra.

negativa, su puntuación se invierte. Por ejemplo, si encontramos la frase "No me gusta el helado", clásicamente se consideraría que está presente la palabra "gusta" como positiva, por lo que la puntuación sería +1. Sin embargo, a partir de 2004, Hu y Liu, considerarían que el "no" que precede a la palabra "gusta" invierte la carga y la frase se convierte en negativa.

- El segundo enfoque, con investigadores como Polanyi y Zaenen (2006), optan por definir unas medidas concretas a partir de *cuánto* consideran que la expresión negativa afecta a la positiva. También consideraron que hay más factores que influyen en dirección de la polaridad, como es el caso de los *intensificadores*. Por ejemplo, no es lo mismo "me gusta el helado" que "me gusta mucho el helado". A partir de este enfoque se han desarrollado diferentes formas de puntuar la polaridad, muchas veces ampliando el rango de los valores. En el caso de Taboada el al. (2011), el rango fue ampliado desde -5, totalmente negativo, hasta +5, totalmente positivo.

8.1 Formulación del cálculo

Veamos, a continuación, lo que calcula la función `polarity`:

- Primeramente, se detectan las palabras positivas y negativas a partir del diccionario que se haya aplicado.

- Para cada palabra se determina un rango de palabras que pueden influir en su puntuación o dirección de polaridad, por ejemplo, se pueden tomar 5 palabras a su alrededor hacia ambos sentidos. De esta forma se determina un cluster para cada término, denominado *grupo de contexto*.

- A cada uno de los términos del grupo de contexto se le asigna una puntuación neutra, negadora o intensificadora, con el fin de poder entender cómo es la carga de la palabra influida por las demás. Se ha de decir que las palabras negadoras pueden invertir tanto la puntuación de una palabra positiva, como negativa.

  Los términos del grupo de contexto se denotan de la siguiente forma:

- $\mathbf{X}_i = \{X_{ij} : j = 1, \ldots, n_i\}$ del grupo de contexto de la palabra i (positiva o negativa)

- $A_i = \sum_{j=1}^{n_i} \mathbf{1}_A(X_{ij})$, número de palabras amplificadoras.

- $D_i = \sum_{j=1}^{n_i} \mathbf{1}_D(X_{ij})$, número de palabras disminuidoras.

- $N_i = \sum_{j=1}^{n_i} \mathbf{1}_N(X_{ij})$, número de palabras negadoras.

  donde

$\mathbf{1}_A$, $\mathbf{1}_D$ e $\mathbf{1}_N$ son las correspondientes funciones indicadoras (toman el valor 1, si el término pertenece a algún tipo de palabras anteriormente indicadas, o bien 0 si no pertenece).

- Las palabras positivas y negativas se puntúan como +1 o -1, pudiendo modificarse esta puntuación.Por su lado, las palabras neutrales no afectan esta ecuación, aunque sí tienen un efecto sobre el recuento de palabras. Los intensificadores tienen unos pesos definidos en cuanto a lo que modifican la carga emocional.

Se denota por W_i el peso de la palabra polarizada i:

$$W_i = \begin{cases} +1 & \text{si } i \text{ positiva} \\ -1 & \text{si } i \text{ negativa} \end{cases}$$

`y se calculan los valores:`

$$W_i^{neg} = N_i \ mod 2$$

$$Dism = max(-c \cdot (D_i + W_i^{neg} \cdot A_i), -1)$$

$$Amp = c \cdot A_i \cdot (1 - W_i^{neg})$$

donde c es el peso de los intensificadores.

- A continuación, se obtiene un total de la polaridad para el cluster teniendo en cuenta el efecto de los modificadores.

$$C_i = (1 + (Dism + Amp)) \cdot W_i \cdot (-1)^{N_i}$$

- Para saber la puntuación final de la cadena de texto, estas puntuaciones de los grupos de contexto son sumados y divididas por la raíz cuadrada del total de las palabras que la conforman:

$$C = \frac{\sum C_i}{\sqrt{n}}$$

En este sentido, la función `polarity` ofrece una mezcla de los dos enfoques en el sentido de que considera la inversión de los valores positivos cuando se encuentran precedidos de *negadores*, así como la influencia de los *amplificadores* y los *disminuidores*.

8.2 Preparación de los diccionarios

Resulta necesario llevar a cabo una preparación de los diccionarios porque el paquete está diseñado para inglés, aunque ofrece la posibilidad de elegir diccionarios diferentes. En este sentido, aprovechando este punto, se puede importar un diccionario para otro idioma y la función será capaz de procesar el texto en función de este.

Para el presente trabajo se utiliza el diccionario para español ofrecido por Molina-González et al. (2013), llamado iSOL [2]. Este diccionario reune 2509 palabras definidas como positivas y 5626 palabras negativas, puntuadas como +1 y -1 respectivamente. Este conjunto de términos se ha generado a partir de una traducción automática de un diccionario elaborado por Bing Liu [3], aunque revisado y ampliado.

```
diccionario <- read.csv2('../isol_completo.csv', header = T)
head(diccionario)
```

```
##       Palabras Puntuaciones
## 1 abiertamente            1
## 2       abrazo            1
## 3      abrazos            1
## 4   absorbente            1
## 5  absorbentes            1
## 6     absuelta            1
```

Por otro lado, tener el diccionario de palabras positivas y negativas no cubre todas las necesidades que tenemos en la formulación aplicada porque carece de amplificadores,

[2] El diccionario se puede descargar en el siguiente enlace: http://timm.ujaen.es/recursos/isol/

[3] El diccionario original se puede encontrar en la siguiente página web: http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html.

disminuidores y negadores, por lo que su determinación ha de ser propia. Así, se ha llevado a cabo una búsqueda en la Real Academia Española con el fin de encontrar términos que se ajustasen a las necesidades del estudio.

Finalmente, se han creado tres listas: 50 amplificadores, 6 negadores y 13 diminuidores. Se ha de decir, por su lado, que no se trata de una determinación exhaustiva, ni de una definición teórica, más allá de las pruebas de las posibilidades de las herramientas. Por lo tanto, no se trata de afirmar que estas palabras sean exclusivas ni únicas de utilidad en este contexto.

```
# Visualizamos 10 términos dentro de cada categoría
amplificadores[1:10]
```

```
##  [1] "extremadamente" "cierto"         "enorme"         "enormemente"
##  [5] "extremo"        "muy"
##  [7] "altamente"      "enormemente"    "inmensamente"   " incalculable"
```

```
negadores[1:6]
```

```
## [1] "no"      "sin"     "nunca"   "jamas"   "tampoco" "ni"
```

```
diminuidores[1:10]
```

```
##  [1] "apenas"        "relativamente" "debilmente"    "poco"
##  [5]   "poca"          "solamente"     "menos"
##  [8] "casi"          "ridiculamente" "inferior"
```

8.3 Aplicación de 'polarity'

En primer lugar, cargamos el archivo de datos creado a partir del preprocesamiento y cargamos la librería (o paquete) necesario.

```
load('df_procesado.RData')
library(qdap) # Contiene 'polarity'
```

En primer lugar, al aplicar esta función a una variable de texto directamente, nos damos cuenta de que calcula un único valor para todo el conjunto. Esto no resulta conveniente pues lo que nos interesa conocer es la puntuación de polaridad de cada una de las críticas

de cine.

Por este motivo, para que se calcule la polaridad para cada valor o conjunto de texto dentro de la variable, es necesario desarrollar una función que, en forma de bucle, lo aplique en cada caso. Así pues, creamos un objeto vacío que va a contener la información resultante y aplicamos la función `polarity` dentro de otra función creada con la ayuda del bucle `for`.

Como se puede observar, siempre que aplicamos esta función, indicamos cuáles son los objetos en los que se encuentran todos los diccionarios que son necesarios para su ejecución.

```
res_polaridad_sumario <- c() # Objeto vacío

for (i in 1:length(df_polarity$new_sumario)){
  res_polaridad_sumario[i] <- polarity(
    df_polarity$new_sumario[i],
    polarity.frame = diccionario,
    negators = negadores,
    amplifiers = amplificadores,
    deamplifiers = diminuidores)}
```

En cambio, esto no es suficiente para obtener el resultado de la polaridad como tal porque la función nos devuelve varias salidas para cada elemento. Si se quiere procesar posteriormente o simplemente obtener las puntuaciones por separado, lo que simplifica la variable resultante, es necesario crear otro bucle para su extracción.

```
  # Visualizamos el resultado para el primer texto
res_polaridad_sumario[[1]]
```

```
##   all wc  polarity pos.words neg.words          text.var
## 1 all  3 0.5773503     amigo         - may quieres amigo
```

```
pol_sumario<- c() # Objeto vacío

for (i in 1:length(res_polaridad_sumario)){
  pol_sumario[i] <- res_polaridad_sumario[[i]]$polarity}
```

Para comprobar que la variable se ha extraído correctamente, podemos recurrir a la función `str`, ya que ofrece un resumen de sus distintas características.

```
str(pol_sumario); length(pol_sumario)
```

```
##  num [1:3878] 0.5774 0 0 0.0816 0.2673 ...
```

```
## [1] 3878
```

Ahora bien, aplicamos el procedimiento del mismo modo a la variable `new_texto` y observamos el resultado que se obtiene con la identificación de las palabras positivas y negativas, así como la puntuación de polaridad.

```
# Calculamos las puntuaciones para cada caso
res_polaridad_texto <- c() # Objeto vacío

for (i in 1:length(df_polarity$new_texto)){
  res_polaridad_texto[i] <- polarity(
    df_polarity$new_texto[i],
    polarity.frame = diccionario,
    negators = negadores,
    amplifiers = amplificadores,
    deamplifiers = diminuidores)}

#Creamos una función que guarde las puntuaciones de polaridad
pol_texto <- c() # Objeto vacío

for (i in 1:length(res_polaridad_texto)){
  pol_texto[i] <- res_polaridad_texto[[i]]$polarity}

# Comprobamos la correcta creación
str(pol_texto); length(pol_texto)
```

```
##  num [1:3878] 0.429 -0.555 -0.773 -0.147 1.424 ...
```

```
## [1] 3878
```

A continuación, vamos a ver el resultado obtenido a partir de la aplicación de `polarity`en un elemento concreto dentro de la variable `new_texto`. Así, se puede ver que obtenemos un `data.frame` o un conjunto de variables en el mismo objeto, dentro del cual se da el

resultado de 6 variables concretas:

- La primera variable es la selección que se lleva a cabo dentro del texto, dentro de la cual se establece la opción de `todo el contenido`, por lo que la salida indica `chr "all"` o bien "todos los contenidos de tipo caracter".
- En segundo lugar se indica la cantidad total de palabras, que, en este caso, es de 266 `wc`.
- La tercera variable `polarity` es de caracter numérico `num` y contiene la puntuación de polaridad asignada al mensaje.
- La cuarta variable es una lista de tipo `chr` o caracteres, que incluye las palabras positivas identificadas.
- La quinta variable es una lista `chr`que contiene las palabras negativas identificadas.
- Por último, la sexta variable es el texto que se ha analizado.

```
str(res_polaridad_texto[[1]], width = 70, strict.width = 'wrap')
```

```
## 'data.frame':    1 obs. of  6 variables:
## $ all : chr "all"
## $ wc : int 266
## $ polarity : num 0.429
## $ pos.words:List of 1
## ..$ : chr "amigo" "pacifica" "amigo" "prima" ...
## $ neg.words:List of 1
## ..$ : chr "terror" "cara" "terror" "locura" ...
## $ text.var : chr "may quieres amigo peliculas recuerdan terror
##    siempre lleva garras acero mano mascara cara terror locura
##    encuent"| __truncated__
```

Finalmente, podemos ver la distribución de las polaridades que se ha obtenido para las dos variables de texto. Se ha de indicar que la función `polarity` permite fijar el rango de la polaridad entre +1 y -1, en cambio, con el fin de observar cuál sería la complejidad encontrada, este parámetro no se ha fijado.

Así, podemos ver en la distribución de frecuencias de la variable `texto` que la mayor concentración de los casos, como es lógico, se sitúa alrededor de 0 y, aparentemente, teniendo en cuenta las puntuaciones a partir de 0,5 hacia ambos lados de la distribución, predominan las puntuaciones positivas.

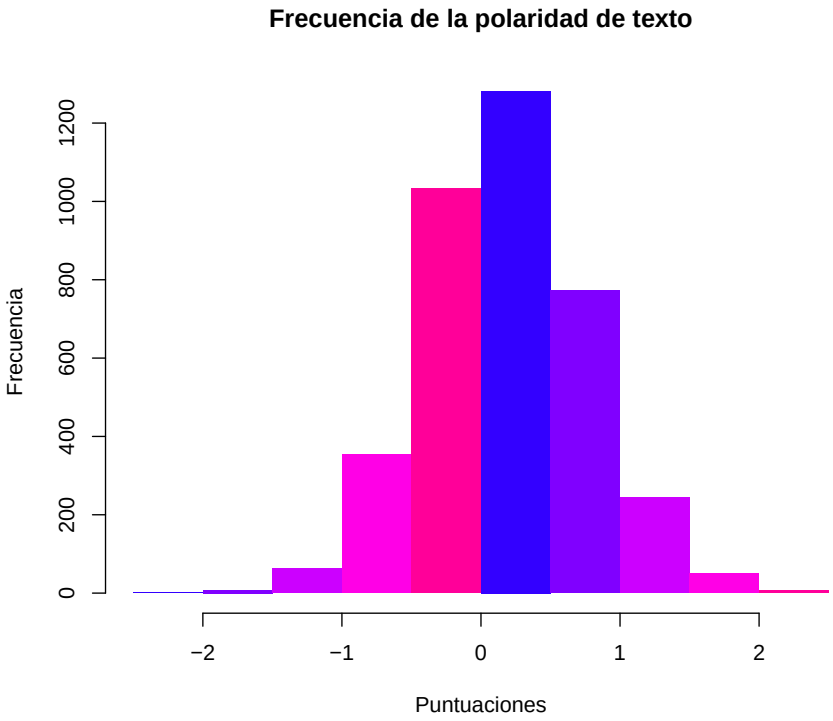

Figure 8.1: Frecuencia absoluta de la polaridad de 'texto'

En lo que se refiere a la variable sumario, se produce el mismo efecto, siendo mayor la concentración para el caso de las valoraciones positivas. En cambio, se hace notorio el efecto de algunos casos que podrían ser atípicamente negativos, puesto que hay algunas puntuaciones que llegar a ser -3.

8.4 Combinación de las bases de datos

Por último, como siempre que se modifican o crean variables nuevas, llevamos a cabo la práctica de volver a combinar los datos de forma que la base de datos final esté constituida y la investigación sea totalmente replicable y posible de continuar, sin tener que volver a pasar por el proceso descrito.

```
load('df_dtm_sum.RData') # Sumario


df_tm_sum <- df_dtm_sum
colnames(df_tm_sum)
```

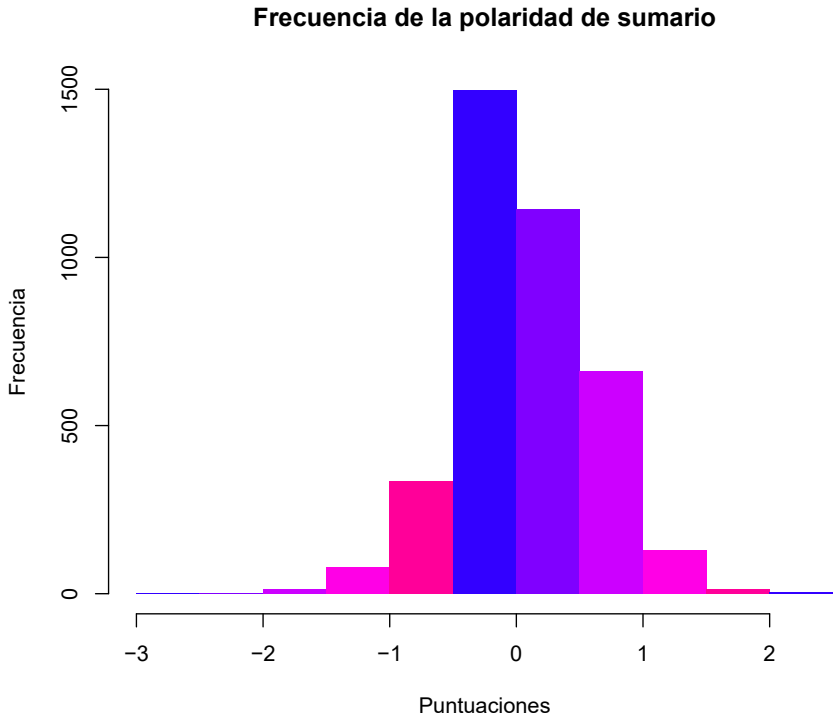

Figure 8.2: Frecuencia absoluta de la polaridad de 'sumario'

```
# Se añade como nueva variable la valoración de polaridad de 'new_sumario'
# a la matriz de términos DTM de sumario
df_tm_sum$pol_sumario2 <- pol_sumario
head(df_tm_sum$pol_sumario2)

# Se añade como nueva variable la valoración de polaridad de 'new_texto'
# a la matriz de términos DTM
load('df_dtm_text.RData') # Texto

df_tm_text <- df_dtm_text
df_tm_text$pol_texto2 <- pol_texto

# Por último, a las dos bases de datos resultantes se ha de añadir la 'puntuación'
# cada por cada usuario en su comentario con el fin de poder llevar a cabo
# análisis posteriores
puntuacion2 <- df_polarity$df.puntuacion
```

```r
# La puntuación para todas las variables es la misma, siempre que se
# corresponda con el número de comentario
df_tm_sum$puntuacion <- puntuacion2
df_tm_text$puntuacion <- puntuacion2

save(df_tm_sum, file = 'df_tm_sum_completo.RData')
save(df_tm_text, file = 'df_tm_text_completo.RData')
```

Capítulo 9

MÉTODOS DE CLASIFICACIÓN

Con el fin de mostrar algunos de los análisis en los que es posible aplicar la variable `polaridad`, se ofrece un ejemplo de técnicas de clasificación para datos de texto: *árboles de decisión* y *bosques aleatorios*. [1]. No se trata de una aplicación del análisis de sentimientos como tal, pero sirve para dar una idea de las múltiples formas de aprovechar los datos de texto.

Para llevar a cabo estos procedimientos, se parte de los datos recogidos y procesados en la `matriz de documentos términos` con la ayuda del paquete `tm`. La variable `puntuación` está recogida dos veces, en su estado original y, por otro lado, con una recodificación en dos categorías, cuando es mayor de 3 equivale a 1 y 0 en otro caso, para facilitar la conformación de los árboles de decisión y su interpretación.

En este capítulo se llevará a cabo un análisis de la variable `sumario`, sin estudiar la variable `texto` [2]. Esta decisión ha tenido que ser tomada debido al carácter ilustrativo del capítulo y, por otro lado, a la carga computacional que supone su procesamiento.

Ahora bien, si bien en todo el documento hemos hablado del análisis de datos no supervisados, al incluir una variable respuesta pasamos a las ténicas supervisadas de clasificación, puesto que es posible comprobar la bondad de ajuste del clasificador obtenido.

La forma habitual de tratar los datos supervisados consiste en distribuir los casos recogidos

[1] El código utilizado en el presente capítulo se encuentra en la carpeta de Dropbox del siguiente enlace: https://www.dropbox.com/sh/i0qm1ao7izxm69c/AADVgvyBmYI4jDDMgj8dh9Kla?dl=0

[2] No se van a tratar las variables originales, sino las extraídas a partir de las frecuencias con la `matriz de documentos término`.

en la base en una serie de subconjuntos. En este caso, se establecerán los grupos de entrenamiento o *training* (80%) y de prueba o *test* (20%), siendo el primero a partir del cual se ajusta la clasificación del segundo. Con este procedimiento se estima la precisión de las predicciones.

9.1 Árboles de decisión

El árbol de decisión es una técnica que clasifica los datos extrayendo particiones binarias de forma recursiva, es decir, cada bifurcación se basa en un único criterio determinado y, siguiendo el planteamiento aplicado en este apartado, tiene dos opciones a la hora de ser clasificadas, puesto que la *respuesta* es dicotómica: película "buena" o película "mala".

Principalmente, hay dos formas de visualizar los resultados: en forma de texto y como gráfico de árbol. Así, para el propósito ilustrativo de los resultados que se obtienen con este tipo de técnicas, nos centraremos exclusivamente en la expresión gráfica.

9.1.1 Árbol de decisión sin `polarity`

Cuando se entrena un árbol de decisión, este produciría clasificaciones hasta que no hubiese más cambios posibles. Sin embargo, en la práctica esto supone que una pérdida de tiempo y de eficiencia. Así, para que el árbol crezca únicamente con las clasificaciones más relevantes, se emplea el criterio de *complejidad*. Cuando el criterio de complejidad es 0, el árbol crece al máximo, y se suele tener en cuenta un valor estándar, que es 0,001.

Así, a continuación, vemos que hay dos *nodos* que clasifican los documentos en dos categorías de opinión sobre la película en cuanto a su puntuación: 0 significa que la película fue clasificada como "mala" y 1 que fue clasificada como "buena", por ponerle etiquetas.

Por lo tanto, el primer nodo parte de la palabra *obra*. Si un comentario tiene una puntuación de frecuencia de esta palabra menor que 1, habrá puntuado que la película es "mala" con un 66% de probabilidad, quedando un 34% de probabilidad de haberla puntuado como "buena".

Además, el segundo nodo indica que si un texto tiene una puntuación de *obra* por debajo de 1 y, a su vez, su puntuación de *mejores* está por debajo de 1, habrá puntuado la película

como “mala” con una probabilidad del 67%. Por otro lado, si tiene una puntuación de *mejores* mayor que 1, tiene una probabilidad del 66% de haberla considerado “buena”.

Si el comentario tiene más de 1 en *obra* [3], automáticamente pasará a a tener una probabilidad de haber calificado la película como “buena” con una probabilidad del 66%.

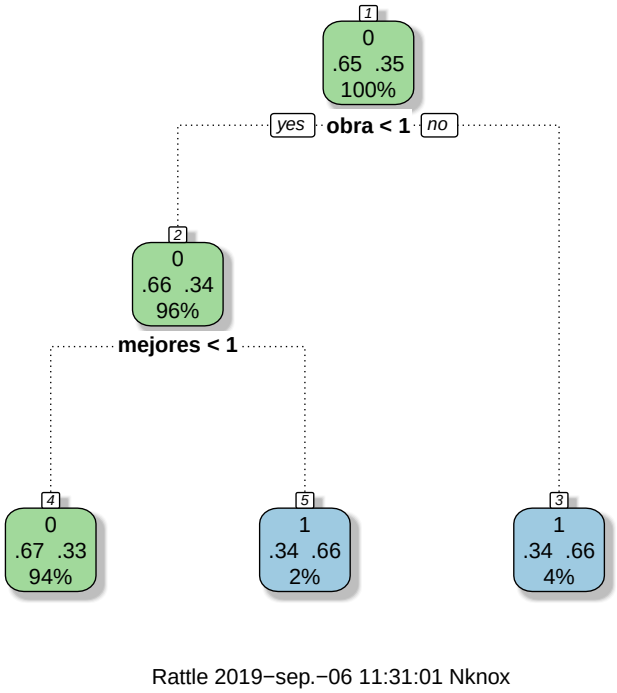

Figure 9.1: Árbol de decisión sin polaridad

9.1.2 Árbol de decisión con polarity‘

Si incluimos la variable `polaridad`, esta va a tomar el lugar del principal criterio clasificador. Así, si su valor es menor que 0,25, la puntuación de la película será “mala” con una probabilidad del 72%. En cambio, si es superior a ese límite, otra vez la palabra que más determina es *obra*, cuyo valor si es mayor que 1, clasifica a la película como “buena” con una probabilidad del 72%.

[3]Es interesante que los resultados obtenidos en las correlaciones adquieren sentido en cuanto a que la palabra *obra* sea el clasificador más importante y este era el término que más correlacionado esta con la palabra *maestra*.

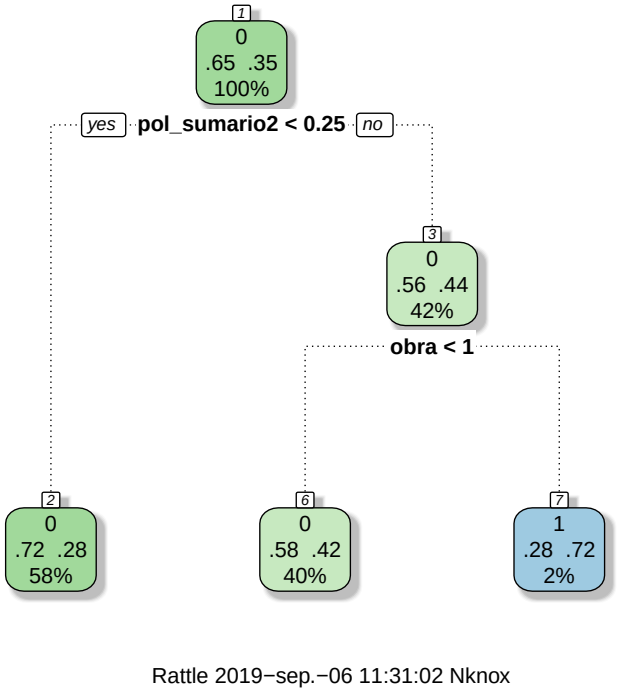

Figure 9.2: Árbol de decisión con polaridad

9.2 Bosque aleatorio

En cuanto al bosque aleatorio, se trata de un procedimiento que prueba diferentes opciones posibles de los árboles, por lo que se constituye como si fuese su continuación. El procedimiento se hace con la ayuda de *bagging*, que es una selección aleatoria de casos con reemplazamiento.

9.2.1 Bosque aleatorio con `polarity`

Se construyen 500 árboles con el valor por defecto y en el modelo fueron utilizados 13 clasificadores. Asimismo, como se puede ver, se muestra una matriz de confusión en cuanto a los aciertos que se han dado en cada categoría.

De este modo, se puede ver que han sido considerablemente mejor clasificados aquellos casos que puntuaban la película como "mala": 1452 casos bien clasificados, frente a 313 mal clasificados, teniendo una tasa de error de 17,73%, mientras que aquellos casos que puntuaban la película como "buena", tuvieron una tasa de error del 64,38%.

```
##
## Call:
##  randomForest(formula = as.factor(niv_punt_sum) ~ ., data = crs$dataset[crs$train,
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 13
##
##         OOB estimate of  error rate: 34.05%
## Confusion matrix:
##      0   1 class.error
## 0 1452 313   0.1773371
## 1  611 338   0.6438356
```

Además de esta información, se pueden ver las variables por su importancia en cuanto a la clasificación. En este caso, vamos a mostrar únicamente las 6 primeras, dentro de las cuales están `polaridad`, *obra*, *buena*, *mejores*, *hacia* [4] y *año*.

```
##
## Call:
## roc.default(response = crs$rf$y, predictor = as.numeric(crs$rf$predicted))
##
## Data: as.numeric(crs$rf$predicted) in 1765 controls (crs$rf$y 0) < 949 cases (crs$rf$y
## Area under the curve: 0.5894

## 95% CI: 0.5718-0.6071 (DeLong)

##                   0     1 MeanDecreaseAccuracy MeanDecreaseGini
## pol_sumario2 28.25  8.61                28.56            61.73
## obra         20.77 13.97                25.74             8.71
## buena        17.18 -0.78                14.44             5.31
## mejores      13.79  3.44                13.41             5.36
## hacia         8.20  9.47                11.58             3.02
## año           6.75  9.34                11.32             4.66
```

[4] Este término parece ser un ejemplo de aquellas stopwords que no están incluidas en la depuración automática y que no tiene significado de por sí, pero sí una alta frecuencia.

9.2.2 Bosque aleatorio sin `polarity`

Si ignoramos la variable de polaridad, se emplea el mismo número de clasificadores y la predicción de las películas calificadas como "buenas" empeora, siendo 66,17% de error frente a un 64,38% anterior.

Las variables más importantes adquieren cierta correspondencia con los resultados que se obtenían en el árbol de decisión al no tener en cuenta la polaridad. Asimismo, las tasas de error permanecen, como indicábamos, de forma semejante a las anteriores.

```
##
## Call:
##  randomForest(formula = as.factor(niv_punt_sum) ~ ., data = crs$dataset[crs$train,      c
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 13
##
##         OOB estimate of  error rate: 34.05%
## Confusion matrix:
##      0   1 class.error
## 0 1452 313   0.1773371
## 1  611 338   0.6438356
```

```
##
## Call:
## roc.default(response = crs$rf$y, predictor = as.numeric(crs$rf$predicted))
##
## Data: as.numeric(crs$rf$predicted) in 1765 controls (crs$rf$y 0) < 949 cases (crs$rf$y 1)
## Area under the curve: 0.5894
```

```
## 95% CI: 0.5718-0.6071 (DeLong)
```

```
##                   0     1 MeanDecreaseAccuracy MeanDecreaseGini
## pol_sumario2 28.25  8.61                28.56            61.73
## obra         20.77 13.97                25.74             8.71
## buena        17.18 -0.78                14.44             5.31
## mejores      13.79  3.44                13.41             5.36
```

```
## hacia          8.20  9.47              11.58             3.02
## año            6.75  9.34              11.32             4.66
```

9.3 Balance de los resultados

Los datos de texto pueden ser estudiados de forma cuantitativa de diversas formas, como hemos ido viendo. En cambio, el papel de la carga emocional, por así decirlo, que ofrece la función `polarity` no parece ser lo suficientemente significativa tal y como está formulada en el momento actual.

Viendo los resultados, tanto de los árboles de decisión, como de los bosques aleatorios, los errores de la clasificación tienen pautas muy semejantes con y sin la polaridad. Aunque, bien es cierto, que sería interesante también llevar a cabo el mismo procedimiento para la variable `texto`. Esta contiene más información y puede establecer una mejor estimación, pero computacionalmente su procesamiento es dificultoso.

Capítulo 10

UNA MEJORA DE CARÁCTER INTERDISCIPLINAR

Una vez llevada a cabo la aplicación de las técnicas de procesamiento de texto y de análisis de sentimientos, a partir del proceso de aprendizaje del presente trabajo es posible ofrecer una serie de conclusiones en cuanto al futuro desarrollo de las herramientas estudiadas.

A continuación, se expondrá una serie de aspectos a desarrollar que, como conclusión del análisis llevado a cabo, podrían mejorar significativamente la aplicabilidad y la expansión de la utilización del análisis de sentimientos por las diferentes ramas del saber.

10.1 Desde la Computación

En primer lugar, una parte crucial en el desarrollo del trabajo ha sido el acercamiento a los aspectos computacionales del análisis de sentimientos, entendidos, sobre todo, en relación al uso de código, en vez de interfaz. Llevando a cabo este procedimiento, se puede ver aquellas acciones aplicables directamente y otras que hay que programar manualmente.

Hay determinados aspectos a mejorar que no están directamente relacionados con el análisis de sentimientos, pero que sí afectan a su funcionamiento. A continuación, se expondrán una serie de actuaciones que podrían contribuir significativamente a su mejora, desde la perspectiva del funcionamiento en general:

- Creación de funciones que permitan una lectura automática de los datos desde diferentes formatos de entrada, así como encodings. Esta parte ha supuesto parones significativos en el trabajo, puesto que son pocas las herramientas en Rcapaces de importar datos que tengan alguna clase de errores de formulación en ese formato. Es posible contemplar estos problemas y elaborar funciones o bien paquetes que sean capaces de operar con más de un formato y/o encoding a la vez, en caso de ser necesario.
- Determinación de las salidas de resultados deseadas: sería interesante que existiese un abanico de posibilidades para escoger a la hora de aplicar determinadas funciones. Esto es, en las herramientas utilizadas habitualmente el formato de salida, así como los resultados que se obtienen, están prefijados y no se encuentran, al menos de forma documentada, opciones internas configurables para cambiar aspectos como la organización de la salida, las variables que se quieren extraer, etc.
- Evitar o eliminar en la medida de lo posible la dependencia de otros paquetes o librerías de R: los paquetes observados se especializan en la elaboración de una serie de funciones que, por ejemplo, depuran los datos. Sin embargo, estas funciones admiten únicamente determinados tipos de datos de entrada y, muchas veces, no son compatibles con librerías de organización de los datos. En este sentido, un aspecto importante para ser trabajado es la compatibilidad entre paquetes o bien la expansión de las utilidades, con el fin de crear una aplicabilidad más cómoda.
- Automatizar los procesos. Cuando nos encontramos en un entorno sin interfaz, las posibilidades de configuración se amplían significativamente. En este sentido, es posible desarrollar determinados procesos que reproduzcan una serie de pautas llevando a cabo un análisis, con los parámetros deseados a definir, de forma automática desde, por ejemplo, la depuración de los datos o una vez que los datos estén depurados. Una vez que se tenga claro el objetivo del análisis, la automatización permite ahorrar tiempo y esfuerzo, así como hacer el proceso investigador eficiente.
- Revisión de los algoritmos que determinan el funcionamiento interno del flujo del programa. Este aspecto es quizás el más alejado de las posibilidades de cualquier persona que no se dedique plenamente a la informática. Sin embargo, se trata de un aspecto relevante en cuanto a que determinados procedimientos consumen cantidades considerables de memoria interna y de tiempo de computación, lo que dificulta el proceso de investigación. Una reconsideración del funcionamiento interno podría permitir un flujo de acciones eficiente.

Por otro lado, en relación a las utilidades con posible desarrollo para el análisis de sentimientos como tal, podríamos proponer las siguientes actuaciones:

- Extracción de diccionarios temáticos. En el enfoque no supervisado de los datos se utilizan recursos léxicos limitados, que parten principalmente de una serie de grandes diccionarios elaborados por expertos. En cambio, cada temática estudiada y cada conjunto de datos tiene un carácter propio y puede ser de interés conocer el léxico de forma aplicada. En este sentido, es posible crear funciones que extraigan léxico propio de los datos estudiados, lo que permitiría ser capaces de captar los aspectos clave, así como depurar aquella terminología que no aporta significado. Por ejemplo, en el caso de las críticas de cine, se podría prescindir de palabras como "película" o "cine", para centrarnos en aquello que realmente aporte información concreta.
- Elaborar paquetes con más funcionalidades. En primer lugar, ess necesario elaborar funciones destinadas a la determinación e identificación de las emociones que se encuentran en un determinado texto o en un conjunto de ellos. Así, como facilitar el estudio de las emociones entre sí. Esto es, conceder la importancia a las propias emociones y no tanto a las estructuras duales "positivo-negativo".

Por último, relacionado también con la computación, lo lógico es pensar que no todas aquellas personas que quieran utilizar el análisis de sentimientos querrá o podrá llevar a cabo la necesaria introducción al lenguaje de programación utilizado. Por este motivo, sería de gran utilidad plantear una serie de mejoras que puedan introducir a estas personas en las técnicas, sin tener conocimientos informáticos avanzados.

- Creación de interfaz. Una vez que las herramientas sean mejoradas, es posible, sin necesidad de ser informático, crear una interfaz que posibilite la aplicación del análisis sin tener que utilizar código directamente. Actualmente, `RStudio` ofrece la posibilidad de crear interfaces disponibles online con la ayuda de su expansión llamada`Shiny` [1].
- Creación de recursos online para dar soporte técnico y resolver dudas, es decir, la creación de una página web o blog que permita una comunicación directa con los usuarios o que, en el caso de que no se quiera tener una interacción, sirva para la exposición de las posibilidades que ofrece el análisis de sentimientos, así como las utilidades de la posible interfaz. Estas utilidades también pueden ser desarrolladas sin tener amplios conocimientos de diferentes lenguajes de programación con la ayuda

[1] Para más información, consulte la siguiente página web: https://shiny.rstudio.com/

de herramientas como `R Markdown` [2], que permite crear páginas web o bien blogs, con el paquete `blogdown` (Fernández Casal, 2017) [3].

10.2 Desde la Estadística

Hasta ahora únicamente hemos hablado del papel del propio funcionamiento de las herramientas y cómo se puede influir en él. En cambio, el enfoque computacional muchas veces tiene en cuenta únicamente las frecuencias encontradas en los datos de texto o bien estas mismas puestas en relación a un diccionario concreto.

Sin embargo, para dar calidad al análisis de sentimientos es preciso manejar una metodología estadística de fondo que vaya más allá. En este sentido, es preciso dar cuenta a los siguientes aspectos:

- Creación de estimadores. Actualmente, en el enfoque no supervisado, las emociones son medidas en función de las frecuencias de palabras positivas y negativas, relacionadas con sus cargas en cuanto a la influencia de otro tipo de palabras, como es el caso de los negadores. En cambio, el desarrollo de estimadores complejos consistentes en procedimientos estadísticos pueden considerarse inexistentes en este contexto.
- Ponderación de emociones. Una vez que las emociones están identificadas, resulta necesario tener en cuenta aspectos como su naturaleza o intensidad. Esto es, se puede estudiar la positividad o negatividad de una emoción concreta o bien el grado de la emoción. Por ejemplo, en el caso de la vergüenza, podríamos considerar que existen diferentes niveles, dentro de los que se encuentran la timidez, el ridículo, la humillación, etc. [4] El objetivo estaría en determinar en qué nivel se encuentra la emoción expresada. En este sentido, sería posible medir la emoción por sí sola, así como en relación a otras.
- Constitución de una pauta de combinación de unidades de medida de los diferentes diccionarios para que sea posible su aplicación conjunta.

[2] Para más información, consulte: https://rmarkdown.rstudio.com/

[3] Para más información consultar el siguiente enlace: https://bookdown.org/yihui/blogdown/

[4] Siguiendo, por ejemplo, a Gottschalk, A. y Gleser, G. (1969), Manual of instructionforusingtheGottschalk-Gleser Content AnalysisScales, Berkeley, University of California Press.

10.3 Desde la Sociología

Quizás lo más destacable para la sociología está en que, a pesar de no que esta técnica se desarrolla por parte de computación y lingüística, es igualmente necesario partir de alguna teoría de las emociones. En las técnicas de análisis de sentimientos las taxonomías de las emociones utilizadas como base para el desarrollo se denominan *modelos*.

Esto indica que no se trata de una definición de la realidad, sino de una formulación concreta de elementos que dan un determinado resultado, como ocurre cuando hablamos de *modelos* en, por ejemplo, un análisis de regresión. Además, estos modelos proceden de casi exclusivamente de los campos de la psicología y la economía (Miranda, & Guzmán, 2017).

En este sentido, el papel de la sociología, desde la perspectiva de este estudio, se encuentra principalmente en dos aspectos: la aportación de teorías de las emociones, así como su interacción, y la contribución a la cohesión de las distintas ramas que saber que se cruzan en el análisis de sentimientos, a través de la búsqueda de una expresión e interpretación de la metodología de forma que sea comprensible para investigadores de distintas ramas.

Así, las principales actuaciones en el estado actual de la cuestión, son las siguientes:

- El replanteamiento de las emociones como centro de la investigación del análisis de sentimientos, evitando la simplificación polar. Es decir, la Sociología de las Emociones está en una fase de trascender la justificación de la necesidad del análisis de las emociones y de su importancia, sin embargo, se trata de una rama muy concreta y, en el mundo científico en general, este aspecto no está superado. En este sentido, es necesario remarcar la importancia de las propias emociones más allá de su utilización con fines lucrativos.
- El replanteamiento de los diccionarios en castellano, tratando de definir los diferentes grados de las emociones, así como conglomerando las palabras, de modo que sea posible establecer relaciones y obtener una mayor precisión. Esto es, la existencia de los recursos léxicos no ofrece en ningún momento la definición de lo que se entiende por las emociones, por lo que el análisis se convierte en significativamente arbitrario en cuanto a su interpretación.

Por último, como mejora general que influiría en todo el proceso, se puede hablar de la propia participación de los sociólogos en el desarrollo, la utilización y la aplicación de estas técnicas. Es cierto que a veces es necesario abrir el abanico para ser capaces de adaptarse,

pero se compensa con el carácter inherente de la mirada sociológica y por la disposición de muchos sociólogos de introducirse en lo desconocido, como en el caso del presente trabajo, aunque suponga un proceso de aprendizaje de programación.

Capítulo 11

CONCLUSIONES

El objetivo principal de este trabajo ha consistido en llevar a cabo una aproximación hacia las herramientas computacionales y estadísticas desarrolladas para analizar las emociones, concretamente, el análisis de sentimientos con el lenguaje de programación `R`. Esta temática está alejada de la investigación social, debido a un conjunto de razones entre las cuales se pueden encontrar las diferencias de definición terminológica, la separación y desconexión existente entre las ciencias, las limitadas posibilidades de la creación y del desarrollo de herramientas computacionales en relación a la especialización de las profesiones, etc.

En este sentido, el esfuerzo de llevar a cabo este trabajo recae, sobre todo, en tratar de aproximar en última instancia, las ciencias, creando un punto de encuentro y dar luz a un enfoque que se está desarrollando y que puede ser inmensamente útil en los estudios sociológicos.

Resulta de gran interés ser capaces de no solo aportar a la ciencia, sino de combinar los diferentes conocimientos y las diferentes formas de conocer, lo que responde al dicho popular eslavo "una cabeza está bien, pero dos está mejor". Es decir, uniendo los esfuerzos de las diferentes ramas del saber es posible mejorar la producción del conocimiento y su calidad, así como hacer posibles determinadas tareas que manualmente son inabarcables, como, por ejemplo, el procesamiento de grandes cantidades de texto.

El aspecto que se puede suponer como la principal dificultad de la separación entre los científicos de diferentes ramas, en este caso, de computación, estadística y sociología, se encuentra en determinados vacíos de conocimiento que debería ser común. Esto es, resulta

imprescindible que haya personas dispuestas a aprender y a alejarse de los estándares de su propia ciencia con el fin de ser capaces de crear puentes, entendiendo ambas partes.

En este sentido, considero que el proceso de aprendizaje de este trabajo incluye una parte inherente y transversalmente sociológica en cuanto al intento de comprender las mecánicas que subyacen a la formulación de las técnicas de análisis de sentimientos y otra parte fundamental de aprendizaje estadístico y, sobre todo, computacional.

11.1 Objetivos específicos

La presente investigación se establece como el resultado de una inmersión en terreno desconocido, por lo que ha tenido que pasar por diferentes fases de formulación, antes de poder ser constituido. En este sentido, los diferentes objetivos han sido formulados de forma que, primeramente, fuese posible llevar a cabo una aproximación al terreno, antes de ser capaces de aplicar las técnicas de interés.

En cuanto al primer conjunto de objetivos teóricos, se ha descrito lo que se entiende por análisis de sentimientos y se ha contextualizado su posición dentro del análisis de texto en general. Posteriormente se han revisado las herramientas disponibles en `R`, analizando su constitución.

Los objetivos empíricos se han cumplido en cuanto a que se ha utilizado un conjunto de datos real, siendo este procesado y expuesto al proceso de análisis de texto, llevando a cabo sobre él un análisis de polaridad con la función `polarity`.

Por último, quizás el objetivo más complicado para ser cumplido ha sido el estudio de las técnicas existentes para detectar las emociones concretas, puesto que se trata de una clase de herramientas que, en el contexto de la acotación del presente trabajo, se podrían considerar inexistentes.

11.2 Futuras líneas de investigación

El presente trabajo refleja un intento de adaptación de las metodologías clásicas utilizadas en sociología hacia la era de las técnicas del mundo digital, teniendo en cuenta las dificultades que eso conlleva.

En este sentido, las limitaciones que tiene este estudio se encuentran en la propia definición de un estudio exploratorio que se lleva a cabo en un campo desconocido. Se trata de un planteamiento que no toma por objetivo sacar conclusiones sobre una serie de datos concretos, ni de comprender una realidad *per se*, sino de formas de investigar, de entender la investigación y de crear conocimiento.

En este sentido, la continuación natural que se puede plantear es la aplicación de análisis multivariantes a los resultados de los datos transformados a variables numéricas, como, por ejemplo, a partir de la `matriz de documentos término`. A partir de estos datos es posible llevar a cabo técnicas de clasificación y de regresión, tomando como referencia la variable puntuación o bien convirtiendo alguna otra en factor.

En cambio, a pesar de que la falta de la aplicación de técnicas multivariantes puede resultar llamativa, estas no entran dentro de los objetivos propuestos. Lo que resulta verdaderamente relevante de cara al futuro es replantear la polarización de las emociones y la constitución de los diccionarios en el enfoque de los datos no supervisados.

Si fuese posible continuar este trabajo, la vía principal que escogería sería el replanteamiento de la constitución de los diccionarios empleados, el desarrollo de recursos léxicos para castellano, no derivados de inglés, así como el establecimiento de parámetros de depuración del texto según las temáticas necesarias.

BIBLIOGRAFÍA

Almashraee, M., Monett, D., Paschke, A. (2016). Emotion Level Sentiment Analysis: The Affective Opinion Evaluation Emotion Level Sentiment Analysis: The Affective Opinion Evaluation, (May).

Basile, P., Basile, V., Nissim, M., Novielli, N., Patti, V. (2017). Sentiment Analysis of Microblogging Data Sentiment Analysis of Microblogging Data, (January). https://doi.org/10.1007/978-1-4614-7163-9.

Bericat, E. (2000). La sociología de la emoción y la emoción en sociología. Papers, 62, 145-176.

Bericat, E. (2016): Problemas sociales, estructuras afectivas y bienestar emocional (La Catarata).

Betancourt, G. A. (2005). Las máquinas de soporte vectorial (SVMs). Scientia et technica, 1(27).

Castellanos, V., Gómez Yáñez , J. A., y Moraño , X. (2019) Innovación metodológica: R etos de la investigación social en un mundo digital. XIII Congreso Nacional de Sociología.

Coulon, A. (2005), La etnometodología. Madrid: Cátedra.

D'Errico, F., Poggi, I. (2017). Social Emotions. A Challenge for Sentiment Analysis and User Models. Emotions and Personality in Personalized Services.

Faen Scopus, (2015), Funcionalidades avanzadas en Scopus, Elsevier.

Fernández Casal, R. (2017), Creación de sitios web con `blogdown`. https://rubenfcasal.github.io/post/creaci%C3%B3n-de-sitios-web-con-blogdown/.

Hovy, E. H. (2015). What are Sentiment , Affect and Emotion? Applying the Methodology of Michael Zock to Sentiment Analysis, 13–25. https://doi.org/10.1007/978-3-319-08043-7.

Kiritchenko, S., Zhu, X., Cherry, C., Mohammad, S. M., Kiritchenko, S., Zhu, X., ... Mohammad, S. (2014). NRC-Canada-2014: D etecting Aspects and Sentiment in Customer Reviews, (SemEval), 437–442.

Liu, B. (2012). Sentiment Sentiment Analysis Analysis and and Opinion Opinion Mining Mining.

Liu, B. (2015). Sentiment analysis. Mining opinions, Sentiments and Emotions. Illinois, Chicago: Cambridge, University Press.

Miranda, C. H., & Guzmán, J. (2017). A Review of Sentiment Analysis in Spanish, 12(22).

Mur,R. A., Aprendizaje Automático para el Análisis de Datos, Madrid.

Molina-González MD, Martínez-Cámara E, Valdivia M, Teresa M(2015) Crisol: B ase de conocimiento de opiniones para el espanol. Sociedad Española para el Procesamiento del Lenguaje Natural.

Oliden, P. E. (2009). ¿ Existe vida más allá del SPSS? Descubre R. Psicothema, 21(4), 652-655.

Pak, A., Paroubek, P. (2008). Twitter as a Corpus for Sentiment Analysis and Opinion Mining, 1320–1326.

Pang, B., Lee, L. (2008). Opinion Mining and Sentiment Analysis, 2, 1–135. https://doi.org/10.1561/1500000001.

Paradis, E., Ahumada, J. A. (2003). R para Principiantes.

Picard, R. W. (1995). Affective Computing. M.I.T Media Laboratory Perceptual Computing Section Technical Report, 321, 1–16.

Plaza-del-Arco, M. D. M.-G., Jiménez-Zafra, M. T. M.-V. (2018). Lexicon adaptation for spanish emotion mining.

Rosas M. V., Errecalde M. L., Rosso P. (2010) Un framework de Ingeniería del Lenguaje para el Pre-procesado Semántico de Textos. XVI Congreso Argentino de Ciencias de la Computación.

Shevchenko, A. (2017), Vergüenza como institución social. Trabajo de Fin de Grado, Universidad de A Coruña

Sidorov, G. (2014). Creación y evaluación de un diccionario marcado con emociones y ponderado para el español, (Cic). https://doi.org/10.7764/onomazein.29.5

Yihui, X. (2018), R Markdoen: The Definitive Guide. https://bookdown.org/yihui/rmarkdown/

Repositorios

https://www.r-pkg.org/

https://cran.r-project.org/web/packages/

https://cran.r-project.org/manuals.html

Paquetes

Allaire JJ, Cheng J, Xie Y, McPherson J, Chang W, Allen J, Wickham H, Atkins A, Hyndman R, Arslan R (2017) rmarkdown: Dynamic Documents for R. R package version 1.11.

Aria, M. & Cuccurullo, C. (2017) bibliometrix: An R-tool for comprehensive science mapping analysis, Journal of Informetrics, 11(4), pp 959-975, Elsevier.

Duncan Temple Lang and the CRAN Team (2019). XML: Tools for Parsing and Generating XML Within R and S-Plus. R package version 3.98-1.20.

Erich Neuwirth (2014). RColorBrewer: ColorBrewer Palettes. R package version 1.1-2.

Feinerer I, Hornik K (2017) tm: Text Mining Package. R package version 0.7-5.

Feinerer I, Hornik K, Meyer D (2008) Text mining infrastructure in R. Journal of Statistical Software, 25(5): 1-54, ISSN 1548-7660. http://www.jstatsoft.org/v25/i05.

Frank E Harrell Jr, with contributions from Charles Dupont and many others. (2019). Hmisc: Harrell Miscellaneous. R package version 4.2-0.

H. Wickham. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York, 2016.

Hadley Wickham (2017). tidyverse: Easily Install and Load the 'Tidyverse'. R package version 1.2.1. https://CRAN.R-project.org/package=tidyverse.

Hadley Wickham, Jim Hester and Jeroen Ooms (2019). xml2: Parse XML. R package version 1.2.1.

Hadley Wickham, Jim Hester and Romain Francois (2018). readr: Read Rectangular Text Data. R package version 1.3.1.

Kirill Müller and Hadley Wickham (2019). tibble: Simple Data Frames. R package version 2.1.3.

Milan Bouchet-Valat (2019). SnowballC: Snowball Stemmers Based on the C 'libstemmer' UTF-8 Library. R package version 0.6.0.

R Core Team (2019). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria.

Rinker T (2013). qdap: Quantitative discourse analysis package [Computer software manual]. Buffalo, NY: Retrieved from http://github.com/trinker/qdap ((Computer Software, Version 2.2.5))

Stefan Milton Bache and Hadley Wickham (2014). magrittr: A Forward-Pipe Operator for R. R package version 1.5.

Terry Therneau and Beth Atkinson (2019). rpart: Recursive Partitioning and Regression Trees. R package version 4.1-15.

Wickham H, François R, Henry L, Müller K (2018) dplyr: A Grammar of Data Manipulation. R package version 0.7.6.

Williams, G. J. (2011), Data Mining with Rattle and R: The Art of Excavating Data for Knowledge Discovery, Use R!, Springer.

Yihui Xie (2016). bookdown: Authoring Books and Technical Documents with R Markdown. Chapman and Hall/CRC. ISBN 978-1138700109