UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Departamento de Sistemas Informáticos y Computación
Universitat Politècnica de València

Extracción Automática de Categorías en Tuits

Trabajo fin de máster

Máster Universitario en Inteligencia Artificial,
Reconocimiento de Formas e Imagen Digital

Autor: Carlos José Villar Lafuente

Tutores: Miguel Rebollo Pedruelo
Elena Del Val Noguera

Curso 2017-2018

Gracias a mi madre,
Jose, Jesús, y Diego,
todos me han ayudado
de una forma u otra.

Resumen

El presente proyecto aborda la creación de un clasificador para discernir de manera automática de qué temas se están hablando en Twitter. A partir de el algoritmo Latent Dirichlet Allocation se obtienen una serie de agrupaciones de palabras. Sin embargo, no se proporciona el tema asociado a cada grupo de palabras. En este proyecto se propone un clasificador entrenado con Wikipedia para discernir de qué tratan los temas de la salida de LDA. El clasificador se ha aplicado a un *dataset* de Tuits de ciudaddes de EE.UU. para la extracción de las categorías de las que más hablan los usuarios.

Palabras clave: clasificador, lda, wikipedia, temas, tuits

Abstract

The present project addresses the creation of a classifier to automatically discern which topics are being discussed on Twitter. A series of groupings of words are obtained from the Latent Dirichlet Allocation algorithm. However, the theme associated with each group of words is not provided. In this project a classifier trained with Wikipedia is proposed to discern what the topics of the LDA exit are about. The classifier has been applied to a tweet dataset of US cities. for the extraction of the categories that most users talk about.

Key words: classificator, lda, wikipedia, topics, tweets

Índice general

Índice de figuras

Índice de tablas

Índice de código

CAPÍTULO 1
Introducción

Vivimos en una sociedad moderna donde la nuevas tecnologías permiten a los miembros de la sociedad intercomunicarse entre sí, compartiendo información y opiniones a gran velocidad, superando a la prensa tradicional incluso en formatos más inmediatos (radio, televisión e incluso la propia web). Tanto los líderes de grandes empresas tecnológicas como presidentes de gobiernos comparten cada vez más y más información en redes sociales como Twitter, siendo el actual presidente de EE.UU. (a fecha del presente trabajo), Donald Trump [1], un gran aficionado a esta red social donde anuncia importantes decisiones [1]. En los casos de que ocurra algún tipo de evento, la gente publica rápidamente lo que está sucediendo, llegando a ser la propia fuente a partir de la cual medios más tradicionales se hacen eco de la noticia[2].

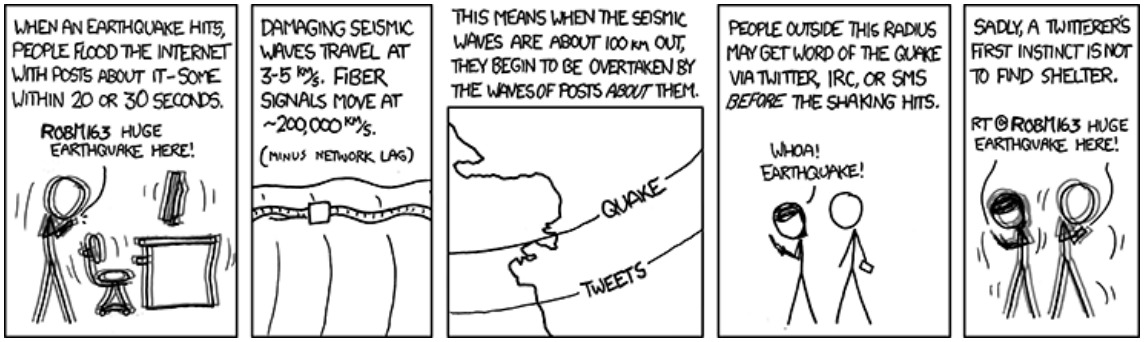

Figura 1.1: Sátira sobre la velocidad de las noticias en Twitter | Randall Munroe

Es por ello por lo que cada vez resulta más interesante el estudio de datos en Twitter, ya sea un análisis sentimental para averiguar la percepción general de las personas hacia una marca/producto/personaje público [3] como para la obtención de datos para la focalización de campañas publicitarias para un determinado sector de la población [4][5][6].

También ha sido utilizado para entender flujos de información[7], patrones de movilidad en las ciudades[8], o como sensores sociales de qué está sucediendo en un determinado momento[9][10].

En este proyecto se utiliza la actividad geoposicionada de los usuarios en Twitter para la extracción de los *topics* más relevantes y la detección automática de categorías. Por medio de este análisis, se puede determinar qué temas están siendo relevantes en una ciudad en un momento determinado. Esta información puede ser utilizada para obtener una visión más realista sobre las necesidades/problemas de los ciudadanos.

[1] https://twitter.com/realdonaldtrump

1.1 Motivación

El análisis de los *topics* o temas sobre los que se habla en redes sociales como Twitter ha sido tratado en múltiples trabajos [11][12][13]. En dichos trabajos se empleaba el algoritmo Latent Dirichlet Allocation (LDA) [14] con el fin de separar las palabras de los distintos tuits en una serie de grupos o temas, de esta forma se puede ver cuáles son las palabras más relevantes en cada uno de los temas.

El problema que presenta LDA, al igual que los algoritmos de agrupamiento (*clustering*), es que su finalidad es procesar un volumen de datos no etiquetados y dividirlos en una serie de grupos que no tiene que corresponderse en número con las clases existentes. Por ejemplo el problema de la flor de iris [15][16], existen tres clases pero se pueden separar los datos en cinco grupos mediante un algoritmo como el k-vecinos, la clasificación posterior indicará la pertenencia a un grupo u otro pero no a una clase u otra *per se*. En este caso LDA no realiza una separación como tal, pero sí que asigna a todos los términos un peso o valor que indica su relevancia en cada una de las categorías.

El propio algoritmo no es capaz de discernir de cuántos temas se está tratando, tiene que ser proporcionado por el usuario o estimado (por ejemplo, calculando la media harmónica). Por consiguiente esto plantea otro problema, se puede sacar un listado de las n palabras más relevantes en el tema k pero desconocemos exactamente de qué está hablando el tema k (deportes, economía, sociedad, etc), si es un número bajo de temas o palabras se puede hacer un etiquetado manual, pero se vuelve impracticable conforme aumenta la cantidad de palabras a analizar y temas a etiquetar. También está el problema de que las palabras han sufrido un procesado para ser reducidas a su raíz (este proceso es conocido como lematización y es más difícil de realizar la tarea de categorización en temas para el usuario.

Todo esto motiva el presente proyecto, la creación de un sistema para asignar de manera automática una categoría a cada uno de los temas que se está tratando en función de la salida de LDA y que ayude al análisis posterior de los datos.

1.2 Objetivos

El objetivo es la creación de un sistema de clasificación en categorías a partir de una serie de palabras con un peso calculado, tal y como obtendríamos a la salida de LDA.

Para alcanzar el objetivo principal del proyecto se proponen los siguientes subobjetivos:

- Implementar la extracción, limpieza y procesado de tuits.
- Implementar un clasificador en categorías entrenado con otra fuente de datos etiquetados.
- Realizar validación cruzada sobre el clasificador.
- Aplicar el clasificador a la salida de LDA.

Al partir de información no etiquetada y proveniente de un medio como es Twitter (dónde los usuario dejan mensajes de longitud corta y no hay un uso adecuado de las normas del lenguaje escrito), se espera que el sistema de clasificación se utilice más como una herramienta como la que pueda ser un traductor automático para un traductor, si bien no lo reemplaza ni ofrece un resultado perfecto, sí que reduce el trabajo y presenta una serie de posibles resultados.

1.3 Estructura de la memoria

El presente documento se va a dividir en las siguientes partes:

Capítulo 2: Estado del arte

En este capítulo se realizará un repaso a los métodos actuales para el análisis de tuits, pero especialmente serán analizados los métodos para la obtención de los temas que está tratando un conjunto de documentos.

Capítulo 3: Modelado

En este capítulo se presentará los distintos algoritmos y técnicas que serán empleadas para la realización del presente trabajo. En este apartado se explicará el algoritmo LDA.

Capítulo 4: Implementación

Se explicará cómo se implementa los algoritmos presentados en el capítulo anterior. Se tratarán las distintas decisiones tomadas así como los problemas encontrados y sus soluciones.

Capítulo 5: Resultados obtenidos

En este capítulo se mostrarán los resultados de aplicar el algoritmo LDA y el clasificador implementado en el capítulo anterior a un conjunto de tuits de distintas ciudades.

Capítulo 6: Conclusiones

En este capítulo se sacarán las conclusiones del presente trabajo, además incluye un apartado con trabajos futuros del proyecto.

CAPÍTULO 2
Estado del arte

En el presente apartado se realizará un repaso a trabajos realizados en el campo del análisis de datos en las ciudades, enfocado principalmente a la extracción de temas que de los que hablan los ciudadanos. También se realiza un repaso sobre trabajos relacionados con la identificación de temas, principalmente en documentos.

2.1 Análisis de información en redes sociales

Existen multitud de formas de analizar y extraer información de los tuits con distintos y variados fines. En el trabajo de M.L. Congosto *et al.*[17] se realiza un análisis de los retuits (*retweet* en inglés, difundir el tuit escrito por otra persona) realizados para discernir afinidades políticas y utilizarlos a modo de sondeo político de cara a las elecciones catalanas de 2011. Además presenta una posible correlación entre el número de tuits asociados a cada partido y el número de votos.

En el artículo de Bodong[12] se realizan una serie de análisis a partir de un conjunto de datos procedente de una serie de conferencias y sus respectivas etiquetas (*hashtags*). Analizando etiquetas, retuits y el propio contenido del mensaje con LDA y otras técnicas, se obtienen datos tales como los temas, grupos de usuarios y sus interacciones o interés cambiante en según que temas tratados en las conferencias.

Otro trabajo es el de Wang [18], en él se presenta un sistema basado en análisis sentimental [19] en el que tras asociar los tuits a cada candidato del ciclo presidencial de 2012 en EE.UU. se intenta averiguar cuál es la opinión de la gente hacia cada uno de los candidatos.

No todos los análisis se realizan sobre el propio mensaje, también hay análisis de los metadatos como la geolocalización con el fin de, por ejemplo, recomendar sitios [20] o de averiguar qué temas tienen mayor relevancia en una determinada ciudad [21][11].

En el trabajo de Liliber Ramos [11], se presenta un análisis estadístico empleando como herramienta el algoritmo LDA [14], el cual tras ser aplicado obtiene un número de temas (fijado previamente tras el uso conjunto de LDA y el cálculo de la media harmónica [22]) y sus n palabras más importantes, todo acotado en espacio en distintas ciudades.

Pereira *et al.* [23] realiza una extracción y minado de datos sobre un conjunto de más nueve millones de tuits de dos grandes ciudades de Brasil. El análisis principal es no supervisado y mediante LDA (aunque existe una parte que si es supervisada). Si bien implementan un etiquetador de los tuits, la clasificación de los temas obtenidos mediante LDA es totalmente manual. Sus resultados muestran similitudes de temas entre ambas ciudades con la variación de aquellos más específicos de cada ciudad.

Grinbegr *et al.* [24] realiza un análisis de los patrones en la vida real a través de los usuarios de *Foursquare*[1], y gracias a su sistema de categorías puede tener los datos ya etiquetados. Es capaz de encontrar los mismos patrones de actividad en Twitter.

Tras examinar los trabajos mencionados en esta sección se ha podido observar que uno de los algoritmos más utilizados para el análisis de temas es el de LDA.

2.2 Identificación de temas

En cuanto al campo de la identificación de temas no es tan abundante como otras ramas del procesado del lenguaje humano, sin embargo existen unos cuantos trabajos de interés.

El trabajo de Hassan *et al.* [25] presenta un sistema en el que se construye un árbol ontológico con la estructura de categorías de la Wikipedia y luego realiza un mapeado entre conceptos presentados en los artículos y temas. El problema que puede presentar este enfoque es que Wikipedia no sigue una estructura en forma de árbol sino de grafo dirigido y existen multitud de ciclos. En este enfoque se tiene en cuenta el contenido del propio artículo (obviando enlaces).

Otro trabajo interesante es el de Chandler May [26], en el cual se estudian diversos métodos para reducir la dimensionalidad de los datos de forma previa a su clasificación.

Hanna M Wallach [27] busca la forma de reducir la memoria y el tiempo de ejecución necesario en un enfoque típico de *bag of words* o de n-gramas infiriendo una serie de hiperparámetros.

Stein *et al.* [28] presentan un *framework* para modelizar el problema de la identificación de temas en documentos. Esquematiza los distintos acercamientos a la identificación de temas y luego analiza el algoritmo empleado por AIsearch[2] (desarrollado por la misma universidad).

Yingjie Lu [29] realiza una clasificación de texto en una comunidad médica en línea con el fin de discernir de qué tema habla cada mensaje. Al tratarse de un dominio específico ha podido crear un conjunto de datos caracterizados con los cuales poder luego clasificar los mensajes.

Schönhofen [30] realiza un clasificador de documentos a partir de la estructura de artículos y categorías de la Wikipedia. Crea un grafo donde los distintos títulos apuntan a los artículos y estos se relacionan con las categorías, de esta forma procesa luego las palabras de los títulos (el contenido del artículo no es utilizado salvo para obtener las categorías) para relacionarlas con las categorías. Con este conjunto de datos puede etiquetar un documento con una categoría de la Wikipedia en función de las palabras que contiene el documento y que estas aparezcan en los títulos de los artículos de la Wikipedia.

Para finalizar, Lin [31] presenta un método para obtener las ideas centrales de un texto, contando los temas de los que habla, identificando y generalizando a través del sistema jerárquico de taxonomía de conceptos que proporciona WordNet.

[1] Aplicación móvil/red social donde el usuario verifica mediante GPS que ha visitado un comercio, según el número de visitas obtiene un estatus y en algunos sitios ofrecen promociones con el fin de fidelizar clientela. También permite a los usuarios encontrar en su zona locales de restauración y/u ocio. `https://foursquare.com/`

[2] `http://www.aisearch.de`

CAPÍTULO 3
Modelado

En este capítulo se procederá a mostrar y explicar los métodos empleados para el desarrollo de este trabajo. Primero se realizará una introducción al proceso general que se va a realizar para acto seguido mostrarla notación que se utilizará a lo largo del trabajo. A continuación se realizará una descripción del algoritmo LDA. Se describirá el clasificador desarrollado para la asignación de temas a partir de la salida proporcionada por el LDA. Finalmente, se procederá a describir el método para la evaluación de los resultados.

3.1 Presentación del modelado

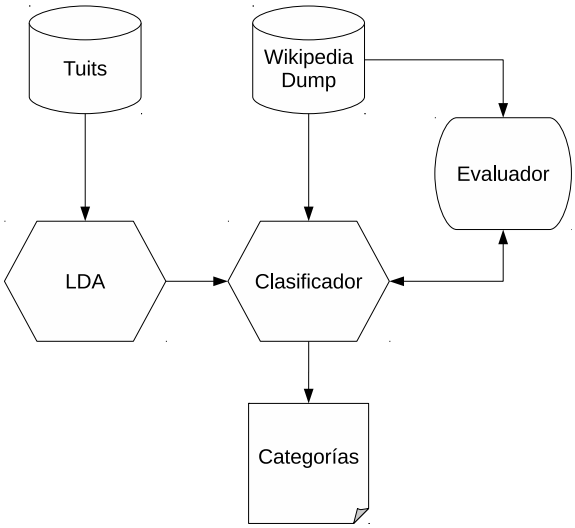

Figura 3.1: Esquema global del trabajo

Como se puede observar en la figura 3.1, disponemos de dos entradas, por un lado están los tuits de las ciudades y por otro la información contenida en la Wikipedia.

Los tuits son procesados mediante el algoritmo LDA, cuya salida pasa a ser una de las entradas del clasificador. El clasificador es entrenado con los datos provenientes de la Wikipedia, empleados a su vez por un evaluador para comprobar el nivel de acierto del clasificador. Una vez entrenado el clasificador, el documento de entrada (la salida de LDA) es procesado y clasificado obteniendo como salida una serie de categorías asignadas como las más representativas del tema hablado en el documento de entrada.

3.2 Notación y terminología

En el presente trabajo se hace referencia a una serie de términos que para ayudar al lector se definen a continuación:

- Palabra: Es la unidad básica de información, por sí sola tiene un significado y se representa como $\boldsymbol{w}$.
- Vocabulario: Es el conjunto de todas las palabras y se representa como $\boldsymbol{v}$.
- Documento: Conjunto estructurado de palabras que expresa una información, se representa como $\boldsymbol{d}$.
- Categoría: Conjunto de palabras que tienen un tema en común, por ejemplo negocios, medicina, informática, etc representada como $\boldsymbol{c}$. También se mencionará como etiqueta de los artículos de Wikipedia.
- Corpus: Conjunto de documentos.
- Corpora: Conjunto de corpus.
- Artículo: Entrada en la Wikipedia, se representa como $\boldsymbol{a}$.
- Título: Título del artículo de Wikipedia, representado por $\boldsymbol{t}$
- Apunta: Diremos que una palabra apunta a un título si $\boldsymbol{w}$ aparece en $\boldsymbol{t}$. Un título apunta a un artículo si $\boldsymbol{t}$ es el título de dicho artículo. $\boldsymbol{a}$ apunta a $\boldsymbol{c}$ (y viceversa) si $\boldsymbol{c}$ es una categoría de $\boldsymbol{a}$.
- Vocabulario de $\boldsymbol{c}$: Palabras que pertenecen a títulos que apuntan artículos pertenecientes a $\boldsymbol{c}$
- $\boldsymbol{w}$ *supports* $\boldsymbol{t}$: Si la palabra $\boldsymbol{w}$ del documento a clasificar $\boldsymbol{d}$ aparece en $\boldsymbol{t}$ y a su vez al menos $n-1$ palabras del título (siendo $\boldsymbol{n}$ el número de palabras en el título) aparecen en dicho documento (por ejemplo puede que sólo aparezca Yeltsin en el documento y el título es Boris Yeltsin). Esta relajación de $n-1$ no se tiene en cuenta si el título sólo consta de una palabra.
- tf_w: Frecuencia del término, número de veces que aparece $\boldsymbol{w}$ en el documento dividido entre el número de palabras totales del documento.
- cf_w: Frecuencia de la categoría, número de categorías que contienen $\boldsymbol{w}$.
- N: Número de artículos de la Wikipedia.
- t_w: Número de títulos que contienen la palabra $\boldsymbol{w}$.
- a_t: Número de artículos apuntados por $\boldsymbol{t}$.
- L_t: Longitud del título $\boldsymbol{t}$ en palabras.
- S_t: Número de palabras del título $\boldsymbol{t}$ en el documento.
- v_c: Numero de *supporting words* de $\boldsymbol{c}$.
- d_w: Valor de *decay* de la palabra $\boldsymbol{w}$.
- B_c: *Supporting words* de $\boldsymbol{c}$, es decir, las *supporting words* de títulos que apuntan a artículos que a su vez apuntan a la categoría $\boldsymbol{c}$.

3.3 Latent Dirichlet Allocation (LDA)

LDA es un modelo probabilístico generativo para colecciones de datos discretos tales como corpora de textos. Se constituye de un modelo Bayesiano en una jerarquía de tres capas en el que cada elemento de la colección es modelado como una mixtura finita sobre un conjunto de temas subyacente. Cada tema es modelado como una mixtura infinita sobre un conjunto subyacente de probabilidades de distintos temas [32].

En los documentos [14] [11] se muestra de forma más detallada el proceso:

LDA asume el siguiente proceso generativo para cada documento $\boldsymbol{d}$ en el corpus $\boldsymbol{D}$:

1. Escoge $N \sim \text{Poisson}(\xi)$
2. Escoge $\theta \sim \text{Dir}(\alpha)$
3. Para cada una de las N palabras de w_n:
   a) Escoge un tema $Z_n \sim$ Multinomial (θ)
   b) Escoge una palabra w_n desde $p(w_n|Z_n, \beta)$, que es la probabilidad multinomial condicionada sobre el tema Z_n

Una variable aleatoria Dirichlet θ k-dimensional, puede tomar valores en el rango de (k-1)-simplex, en dónde simplex es un vector-k dentro de la variable θ aleatoria que se incluye en la expresión (k-1)-simplex si $\theta_i \geq 0, \sum_{i=1}^{k} \theta_1 = 1$, y tiene la siguiente densidad de probabilidad en este simplex:

$$p(\theta|\alpha) = \frac{\Gamma(\sum_{i=1}^{k} \alpha_i)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \theta_1^{\alpha_1 - 1} \dots \theta_k^{\alpha_k - 1} \tag{3.1}$$

En la ecuación 3.1, el parámetro α es un vector-k con componentes $\alpha_i > 0$, y donde $\Gamma(x)$ es la función Gamma. Dados los parámetros α y β, la distribución conjunta de una mezcla de temas θ, un conjunto N de temas z y un conjunto N de palabras w, están compuestas por:

$$p(\theta, z, w|\alpha, \beta) = p(\theta|\alpha) \prod_{n=1}^{N} p(z_n|\theta) p(w_n|z_n, \beta), \tag{3.2}$$

donde $p(z_n|\theta)$ es simplemente θ_i para un único $\mathbf{i}$ tal que $z_n^i = 1$. Integrado sobre θ y sumando sobre $\mathbf{z}$ se obtiene la distribución marginal de un documento:

$$p(w|\alpha, \beta) = \int p(\Theta|\alpha) (\prod_{n=1}^{N} \sum_{z_n} p(z_n|\Theta) p(w_n|z_n, \beta)) d\theta \tag{3.3}$$

Finalmente, tomando el producto de la probabilidad marginal e un solo documento, se tiene la probabilidad marginal de un corpus:

$$p(|\alpha, \beta) = \prod_{d=1}^{M} \int p(\theta_d|\alpha) (\prod_{n=1}^{N_d} \sum_{z_{dn}} p(z_n|\theta_d) p(w_{dn}|z_n, \beta)) d\theta_d \tag{3.4}$$

En el modelado de temas, *topic modeling* (TM) en inglés, un tema es definido como un grupo o *cluster* de palabras, donde cada palabra en el *cluster* tiene una probabilidad de ocurrir para el tema dado. Diferentes temas tienen diferentes *clusters* de palabras

con sus correspondientes probabilidades. Distintos temas pueden compartir palabras y un documento puede contener más de un tema.

LDA es un enfoque de TM donde cada documento es considerado una mixtura de temas y cada palabra en el documento es considerada extraída aleatoriamente de los temas del documento. Los temas son considerados ocultos y deben de descubrirse analizando distribuciones conjuntas computando la distribución condicional de la variables ocultas (los distintos temas).

Puede verse como un algoritmo de agrupamiento (*clustering*) donde los temas corresponden al centro de cada grupo. Temas y documentos coexisten en un espacio de características, donde los vectores de características son vectores de conteo de palabras (*bag of words*). En lugar de estimar un *cluster* usando el enfoque tradicional de la distancia, LDA emplea una función basada en un modelo estadístico de cómo son generados los documentos.

Como salida de este algoritmo tenemos el mismo conjunto de palabras para cada tema, solo que cada una lleva asociado un valor distinto que es la probabilidad de ser generada en cada uno de los temas (por conveniencia se utilizará esta probabilidad como peso de la palabra en una especie de sustitución del tf_w).

Veamos un ejemplo sencillo, tengamos las siguientes frases:

- La nave Enterprise (capitaneada por James T. Kirk) requiere el uso de cristales de dilitio para su correcto funcionamiento.
- Si bien tanto el Maine Coon como el Bosque de Noruega pueden alcanzar grandes tamaños, el Bosque de Noruega no requiere de tanto espacio y puede tenerse en un piso.
- Siempre he preferido la vieja escuela, Kirk, Bones, Scotty, ... frente a Picard, Riker y demás personajes de la nueva generación.
- Un Maine Coon macho adulto puede alcanzar una longitud de un metro (contando la cola) y 8,2 kilos de peso.
- La reacción producida en el motor de antimateria de la Enterprise es controlada mediante dilitio.

Tras la eliminación de las *stopwords* obtendríamos:

- nave Enterprise capitaneada James T Kirk requiere uso cristales dilitio correcto funcionamiento
- Maine Coon Bosque Noruega alcanzar grandes tamaños Bosque Noruega requiere espacio tenerse piso
- Siempre preferido vieja escuela Kirk Bones Scotty frente Picard Riker demás personajes nueva generación
- Maine Coon macho adulto alcanzar longitud metro contando cola kilos peso
- reacción producida motor antimateria Enterprise controlada mediante dilitio

Así, si sacamos las cuatro palabras de mayor relevancia de dos temas podemos tener algo como:

- Tema 1: Kirk, Scotty, Enterprise, dilitio, ...
- Tema 2: Bosque, Noruega, Maine, coon, ...

Este ejemplo es sencillo y podemos discernir la categoría de cada tema:

- Tema 1: Star Trek
- Tema 2: Razas de gato

3.4 Clasificador

Deseamos asignar un conjunto de categorías a un grupo de *topics* de alguna forma, nos encontramos ante una tarea de clasificación y es por ello por lo que debemos construir un clasificador que cumpla dicho fin.

Como el conjunto de datos del que disponemos proveniente de Twitter no está etiquetado de ninguna forma, no podemos entrenar con ellos un clasificador que les asigne etiquetas, como mucho podemos emplear un algoritmo de *clustering* pero ese proceso ya se realiza con LDA. La solución buscada es crear un clasificador que sea entrenado con datos provenientes de otro sitio.

Tras examinar varios artículos vistos en el estado del arte y distintos enfoques (como usar WordNet[1] y/o proyectos similares), el acercamiento empleado al clasificador ha sido el definido en "Identifying Document Topics Using the Wikipedia Category Network" [30]. Wikipedia, sobre todo la versión inglesa, es una gran base de conocimiento cooperativo y libre con más de cinco millones de artículos [2], lo que la convierte en una fuente de datos a tener en cuenta (existen trabajos que hacen uso de la Wikipedia como fuente de información tales como [33][34][35][36]). La creación de un conjunto de datos ya etiquetados es un proceso costoso ya que tiene que haber alguien detrás que lo realice, y es por ello que una fuente colaborativa como Wikipedia resulta de gran interés. Es posible que debido a la diferencia del lenguaje empleado en Twitter y Wikipedia, esta última no sea la más adecuada para entrenar el clasificador, pero al no disponer de un conjunto de tuits etiquetados por categorías y de forma abundante, Wikipedia es la mejor alternativa disponible.

La principal virtud de Wikipedia es también su principal problema, al ser cooperativo no existe una normativa que sea aplicada de forma rigurosa, existiendo artículos en otros idiomas en la propia Wikipedia en inglés. También existen artículos y categorías que tienen una finalidad orientada a la propia gestión de la página y a la organización de los editores web que a proporcionar conocimiento.

A diferencia del trabajo [25], en este no se tiene en cuenta el contenido del artículo, el clasificador es construido a partir de los títulos y categorías de los artículos. También hay que destacar que aquí se han introducido unas pequeñas variaciones respecto al trabajo de [30], tales como no utilizar el *tf* sino el peso que asigna LDA, en este trabajo no se utilizan los artículos en versión *draft*, y la eliminación de ciertas categorías que se han considerado irrelevantes (si bien Schönhofen elimina ciertas categorías, es posible que las categorías consideradas por ambos como irrelevantes puedan diferir).

[1] `https://wordnet.princeton.edu/`
[2] A fecha de 2018-05-14, consultado en `https://stats.wikimedia.org/EN/SummaryEN.htm`

3.4.1. Preparación del corpus de la Wikipedia

El proceso de preparación de los datos del corpus de Wikipedia en inglés empieza por los siguientes puntos:

1. Reducir el corpus a artículos y redirecciones.

2. Realizar eliminación de *stopwords* y reducción a la raíz sobre los títulos de los artículos.

3. Eliminar categorías pertenecientes al mantenimiento y la administración de la Wikipedia (así como otras categorías que no ofrezcan información relevante).

4. Eliminar categorías con menos de 10 o más de 5000 artículos (los primeros no influirán apenas, los segundos influirán demasiado en el resultado final, apareciendo siempre las mismas categorías generales).

De esta forma acabamos con una estructura que contempla sólo títulos, artículos y categorías sin embargo el artículo no es más que un identificador ya que de cara al clasificador sólo interesa el título y la categoría. En la figura 3.2 se puede observar la estructura de la Wikipedia de forma simplificada, donde un artículo puede estar apuntado por varios títulos, un artículo puede apuntar a varias categorías y una categoría ser apuntada por varios artículos.

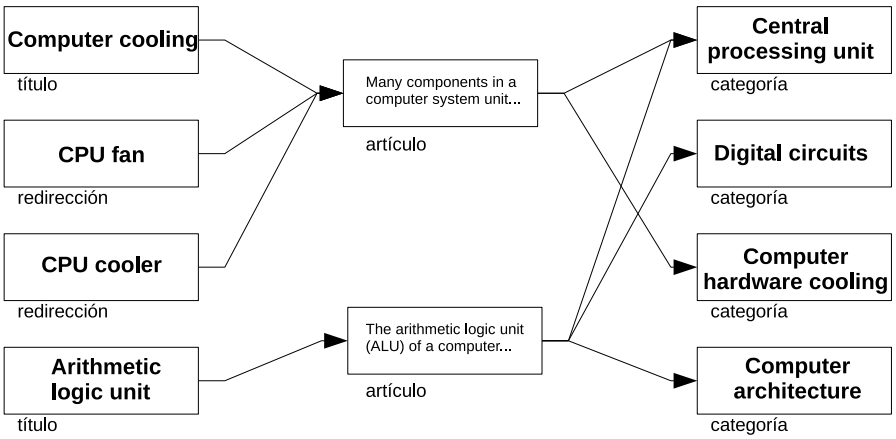

Figura 3.2: Ejemplo estructura de la Wikipedia [30].

Debido al proceso de eliminación de *stopwords* y el *stemming*, tenemos solo palabras relevantes en su formato de raíz, que puede ser compartida por otras palabras. Al final se tiene un conjunto de raíces (de ahora en adelante, palabras) que apuntan a una serie de conjuntos de palabras que forman los títulos. Tras el proceso previo ahora un título (reducido) puede apuntar a más de un artículo. Los artículos pueden ser apuntados por más de un título y apuntan a las categorías.

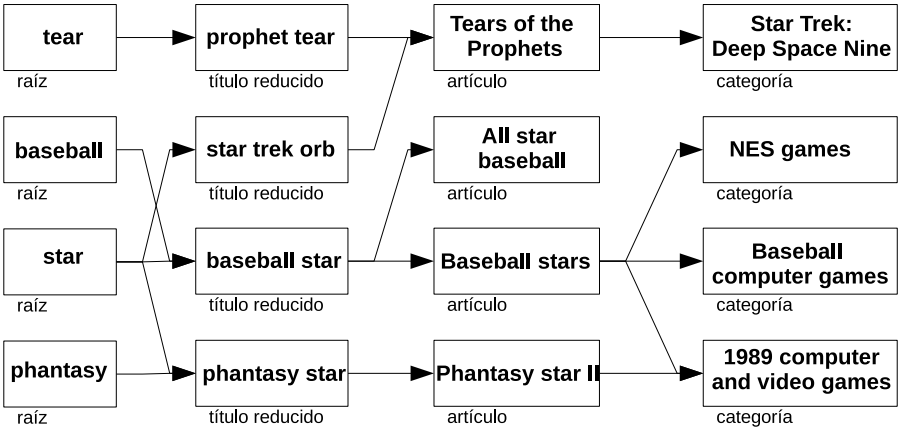

Figura 3.3: Corpus de la Wikipedia preparado [30]

Como se puede observar en la figura 3.3, el artículo "Tears of the Prophets" es apuntado por dos títulos y estos a su vez son apuntados por las palabras que los componen. "Star" apunta a tres títulos distintos, siendo uno de estos "baseball star" que tras la reducción a la raíz de los títulos, ahora apunta a dos artículos totalmente distintos. También se observa cómo una categoría puede ser apuntada por varios artículos y un artículo puede pertenecer a varias categorías.

3.4.2. Identificación de la categoría

El proceso de clasificación (identificación de la categoría) se realiza sacando el peso de cada categoría para el documento a clasificar, en nuestro caso, la salida de LDA siendo cada *topic* un documento.

Pasos a realizar para la identificación del documento

1. Eliminar del documento las *stopwords*, reducir a la raíz y eliminar todas aquellas palabras que no aparecen en los títulos obtenidos de Wikipedia.

2. Recopilar todas las palabras del documento y asignarle un peso:

$$R_w = tf_w \times \log \frac{N}{cf_w} \tag{3.5}$$

3. Recopilar todos aquellos títulos de Wikipedia que tengan todas sus palabras (permitiendo al menos una excepción) en el documento y pesarlos en función de:

$$R_t = \sum_{w \to t} R_w \times \frac{1}{t_w} \times \frac{1}{a_t} \times \frac{S_t}{L_t} \tag{3.6}$$

4. Recopilar los artículos de Wikipedia apuntados por los títulos y pesarlos en función de:

$$R_a = \max_{t \to a} R_t \tag{3.7}$$

5. Recopilar las categorías asignadas a los artículos y pesarlas en función de:

$$R_c = \frac{v_c}{d_c} \times \sum_{a \to c} R_a \tag{3.8}$$

6. Decrementar el peso de las categorías que comparten *supporting words* con otras categorías de mayor R_c

$$R'_c = R_c \times \frac{\sum_{w \in B_c} d_w}{|B_c|} \tag{3.9}$$

$$d'_w = \frac{d_w}{2}, w \in B_c \tag{3.10}$$

7. Seleccionar las categorías de mayor peso.

Tras la limpieza del documento a clasificar (paso uno), se le asigna un peso a cada palabra en función de la frecuencia del término (tf_w) por el número total de categorías dividido por el número de categorías que contienen en su vocabulario dicha palabra w. Como en la salida de LDA sólo aparece una vez cada término acompañado por su peso dentro del tema, se utilizará su peso en lugar de tf_w.

En el paso tres se calculan los pesos de los títulos teniendo en cuenta los pesos de las palabras que los componen.

En el paso cuatro se utiliza la función de máxima ya que tal y como se comentó anteriormente, un artículo puede tener más de un título, por lo que tomamos el mayor peso de los títulos asociados.

En el paso cinco se intenta corregir que destaquen unas categorías sobre otras solo por el hecho de tener un vocabulario mayor. De esta forma se evita que una categoría más específica y relevante se vea eclipsada por otra de menor relevancia pero que al ser más común tenga un vocabulario mayor y por tanto su peso lo sea también ya que engloba a más palabras. Por ejemplo tenemos una categoría c_1 que es muy específica y su vocabulario es muy reducido pero muy específico, en contraposición tenemos una categoría c_2 de amplio vocabulario de palabras más comunes; entonces sea un artículo a que contiene palabras de ambos vocabularios, pero al contener más palabras de c_2 porque son mas comunes, la categoría c_2 tendrá más peso que c_1 aunque sea más específica y el documento realmente pertenezca a c_1.

En el paso seis, todas las palabras empiezan con un valor de *decay* igual a uno y, conforme son procesadas como *supporting words* de categorías de mayor R_c, este valor d_w se va disminuyendo. Finalmente se ordenan las categorías de mayor a menor peso y se escogen las n categorías de mayor R_c.

Finalmente en paso siete, de entre todas las categorías escogemos las n de mayor peso como etiqueta clasificadora.

De esta forma se obtiene una (o unas, dependiendo del n escogido) etiqueta de categoría para el documento de entrada del clasificador. En este proyecto los *topics* de salida del algoritmo LDA son los documentos de entrada del clasificador.

3.5 Evaluación

Para la evaluación del clasificador de *topics*, como no se dispone de un conjunto de tuits procesados mediante LDA y etiquetados, se evaluará clasificando en su lugar artículos de la Wikipedia. Con el fin de no sobre entrenar y obtener unos resultados desvirtuados, se aplicará la técnica de la validación cruzada, que consiste en realizar distintas particiones del conjunto de datos en pares de subconjuntos **entrenamiento** y **test**. La unión de un conjunto de entrenamiento con su respectivo test da como lugar el propio conjunto de datos original, es decir, que el original no se divide en varios pares, sino que los pares son los mismos datos pero en un reparto distinto.

La validación cruzada:

- Serán diez pares de **entrenamiento** y **test** en una proporción 99 % - 1 % (por temas de rendimiento en los test).
- Dos conjuntos de entrenamiento cualesquiera no tendrán de forma exacta todos los artículos entre sí.
- Entrenar el clasificador con el conjunto de **entrenamiento** y clasificar su respectivo **test**.
- Calcular métrica de la clasificación.

Podemos ver un ejemplo simbólico de este proceso en la figura 3.4 donde el *dataset* es replicado y dividido en distintos pares de **entrenamiento** y **test**.

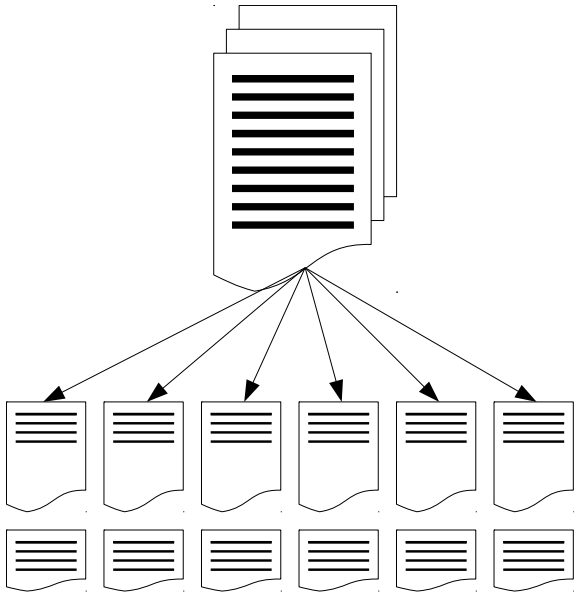

Figura 3.4: Representación de la división para la validación cruzada

Métricas empleadas:

- Precisión (*Precision*): Calculamos cuán acertados han sido los resultados obtenidos mirando cuántas categorías se han acertado.

$$\text{precision} = \frac{|\text{verdaderos positivos}|}{|\text{verdaderos positivos}| + |\text{falsos positivos}|} \tag{3.11}$$

- Exhaustividad (*Recall*): En esta métrica nos interesa saber cuántas categorías se han acertado pero también cuántas de las correctas han sido interpretadas como irrelevantes.

$$\text{recall} = \frac{|\text{verdaderos positivos}|}{|\text{verdaderos positivos}| + |\text{falsos negativos}|} \tag{3.12}$$

- Puntuación F1 (*F1-score*): Esta métrica intenta aunar ambos conceptos, ya que nos interesa tanto la precisión como no rechazar categorías correctas.

$$\text{F1-score} = \frac{2 * (\text{precisión} + \text{exhaustividad})}{\text{precisión} * \text{exhaustividad}} \tag{3.13}$$

3.6 Aleatorización del *dataset*

Antes de dividir los datos para realizar la validación cruzada es recomendable aleatorizar el conjunto de datos para evitar posibles agrupaciones provenientes de la extracción de los datos. Generalmente se disponen en un archivo donde cada línea es un dato, en este caso un artículo, y al aleatorizar las líneas del archivo se aleatoriza también el conjunto de los datos sin desvirtuarlos.

El principal problema de contar con un *dataset* de gran peso como es el caso, es que las limitaciones de memoria principal y secundaria (en lo que a la ubicación de la carpeta */temp* se refiere) impide el uso de las técnicas habituales de aleatorización de las líneas de un archivo (como *shuf* de Linux), teniendo que buscar métodos alternativos que requieran un menor consumo de memoria principal.

Para el caso se ha optado por una técnica que consiste en el empleo de la transformación Schwartziana[37] para indexar las líneas del archivo, en este caso de forma aleatoria, y luego ordenarlas con el fin de realizar la aleatorización. Sin embargo hay que ordenar el archivo y el problema sigue siendo el mismo, para solventarlo se divide en bloques mas pequeños que sean fácilmente manejables y luego se unen.

El proceso de aleatorización consiste en:

1. Aplicar transformación Schwartziana.
2. Dividir el archivo en n bloques.
3. Ordenar cada uno de los n bloques.
4. Aplicar algoritmo de unión en k-vías.
5. Eliminar la información adicional añadida en el paso 1.

3.6.1. Transformación Schwartziana

Esta técnica se utiliza con el fin de agilizar los algoritmos de ordenación reduciendo la complejidad de las comparaciones, este no es el caso pero la técnica empleada sí que resulta útil.

La aplicación de un algoritmo de ordenación implica un gran volumen de operaciones de comparación que si no son triviales pueden acarrear una ejecución mucho más larga. La transformación Schwartziana consiste en precalcular los valores necesarios y añadirlos al propio ítem a ordenar, ordenar en función de esta nueva clave precalculada, agilizando las comparaciones y luego restablecer la información original, con el nuevo orden, eliminando las claves precalculadas. Esto se conoce como *decorate-sort-undecorate*.

Supongamos que tenemos una lista de cadenas de texto `["aaaa", "aa", "a", "aaa"]` y queremos ordenarlas en función de la longitud de cada cadena, en cada comparación entre dos cadenas habría que calcular la longitud de cada una de ellas. Para agilizar se aplica un mapeado de una función que calcula la longitud de cada cadena y se añade a la información, por ejemplo en forma de tupla, quedando como resultado `[(4, "aaaa"), (2, "aa"), (1, "a"), (3, "aaa")]`. Continuando el ejemplo, se ordenaría empleando la nueva información, haciendo comparaciones más sencillas de calcular, `[(1, "a"), (2, "aa"), (3, "aaa"), (4, "aaaa")]` y por último tras eliminar las claves `["a", "aa", "aaa", "aaaa"]`.

3.6.2. Ordenación y unión

La ordenación de los bloques resulta sencilla y no requiere explicación alguna, pero sí el proceso de unión de los distintos bloques, en concreto se realiza una unión en k-vías, manteniendo el nuevo orden. Al no caber todos los bloques en memoria principal hay que tener un *buffer* de lectura de cada uno de los archivos e ir consumiéndolo conforme va avanzando la unión. En la figura 3.5 se puede observar de forma simplificada como una serie de pequeños bloques son unidos en un conjunto mayor.

El proceso de merge en k-vías consiste en:

1. Abrir todos los archivos de los bloques y cargar un pequeño *buffer*.
2. Recorrer todos los *buffers* tomando el primer ítem de cada uno de ellos (recordemos que están ordenados) en busca de cuál es la línea de menor índice.
3. Escribir la línea en el nuevo fichero.
4. Borrar la línea de su correspondiente *buffer* y de ser necesario, cargar más elementos del bloque en este.
5. Volver al paso 2 hasta que todos los *buffers* estén vacíos porque ya no hay más líneas que escribir.
6. Cerrar los archivos tanto los de los bloques como el de salida.

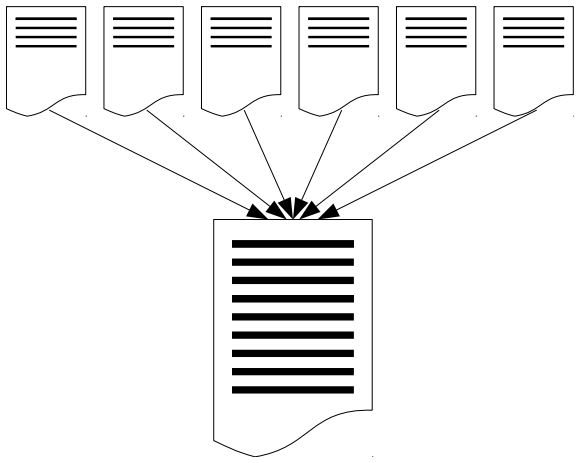

Figura 3.5: Representación de la unión en k-vías

Para agilizar el proceso final, se pueden escribir en el fichero de salida las líneas sin la información de la Schwartziana.

CAPÍTULO 4
Implementación

En este apartado se realizará una explicación de la implementación de los conceptos vistos en **modelado** además de otros procesos que han sido necesarios para la realización del proyecto.

El lenguaje de programación escogido ha sido Python versión tres ya que disponía de algunos conocimientos previos adquiridos a lo largo de la carrera y el máster, siendo la alternativa Java que ofrece peor gestión de la memoria y porque al ser un lenguaje ampliamente utilizado en aplicaciones científicas, es mas fácil encontrar librerías relevantes para el proyecto en Python que en Java.

Todo el código de este proyecto se puede encontrar alojado en un repositorio público de GitHub [1].

4.1 LDA

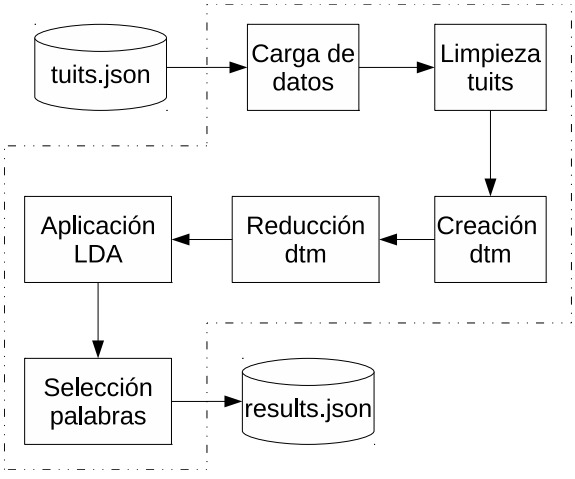

Figura 4.1: Diagrama implementación apartado LDA

En la figura 4.1 se muestra una representación de la implementación del bloque LDA visto en la figura 3.1, concretamente lo que se encuentra dentro de la línea discontinúa es lo que en encierra el bloque mostrado anteriormente como LDA. Primero se cargan los tuits y se limpian, luego se crea una matriz término-documento, acto seguido se reduce

[1] `https://github.com/carlos3dx/tfm2018.git`

eliminando los término de frecuencia inferior a la mediana. Finalmente se aplica el algoritmo LDA propiamente dicho y se guardan las n palabras de mayor peso en cada *topic*.

Para procesar los tuits se ha hecho uso de la librería de LDA de Gensim[2] ya que ofrece una implementación fácil y sencilla de utilizar, concretamente se ha hecho uso de la versión *multicore* para agilizar el proceso de entrenamiento del modelo.

El primer paso cargar los datos y eliminar todo lo que no nos interesa del tuit:

```
def load_tweets(file_path):
    with open(file_path, 'r') as file:
        tweets = json.load(file)

    p_rt_via = re.compile(r'(RT|via)((?:\b\W*@\w+)+)')
    p_user = re.compile(r'@\w+')
    p_punct_digit = re.compile(r'([\W_]+|\d+)')
    p_url = re.compile(r'http\w*\S+')
    p_reduce_whitespaces = re.compile(r'\s{2,}')
    p_whitespaces_removal = re.compile(r'(^\s|\s$)')
    result = []

    stops = set(stopwords.words("english"))
    stemmer = SnowballStemmer("english")
    for tweet in tweets:
        # print(tweet['text'])
        text = tweet['text']
        text = re.sub(p_rt_via, " ", text)
        text = re.sub(p_url, " ", text)
        text = re.sub(p_user, " ", text)
        text = re.sub(p_punct_digit, " ", text)
        text = re.sub(p_reduce_whitespaces, " ", text)
        text = re.sub(p_whitespaces_removal, "", text)
        text = text.lower()
        text_words = text.split(" ")
        text_words = [stemmer.stem(word) for word in text_words if word not
            in stops and len(word) > 0]
        result.append(text_words)

    return result
```

Listing 4.1: Carga y limpieza de tuits

Una vez cargados, se crea un diccionario con las palabras y una matriz donde se relacionan los documentos (tuits) con las palabras que aparecen en ellos y cuántas veces. Con esta información se eliminan las palabras de baja mediana y se recalcula de nuevo el diccionario y la matriz.

```
def main():
    tweets_path = config_lda.get("input_file")
    tweets_clean = load_tweets(tweets_path)

    dictionary = corpora.Dictionary(tweets_clean)
    dtm = [dictionary.doc2bow(text) for text in tweets_clean]
    if reduce:
        tweets_clean = remove_terms_low_median(tweets_clean, dictionary, dtm
            )
        # Calculate again
        dictionary = corpora.Dictionary(tweets_clean)
        dtm = [dictionary.doc2bow(text) for text in tweets_clean]
```

Listing 4.2: Preparación de datos para LDA

[2] https://radimrehurek.com/gensim/models/ldamodel.html

Con la librería de Gensim es muy fácil entrenar un modelo de LDA, tal y como se puede ver a continuación sólo necesita el diccionario, la matriz, el número de topics y cuántas iteraciones del entrenador va a realizar. Con ello sacamos las **n** palabras de mayor peso en cada tema y lo guardamos en un archivo para que luego los temas sean clasificados.

```
    k = config_lda.get("topics")
    ldamodel = gensim.models.LdaMulticore(dtm, num_topics=k, id2word=
        dictionary
                                          , passes=config_lda.get("passes",
                                             20))
    topics_dict = {}
    for topic in ldamodel.show_topics(formatted=False, num_topics=k,
                                      num_words=config_lda.get("words", 10))
                                         :
        topic_words = []
        for pair in topic[1]:
            topic_words.append((pair[0], str(pair[1])))
        topics_dict['topic_' + str(topic[0])] = topic_words

    file_system_json_file = open(config_lda.get("output_file",
                                 "./topics_results.json"), "w")
    file_system_json_file.write(json.dumps(topics_dict))
    file_system_json_file.close()
```

Listing 4.3: Código para LDA

El resultado de salida es en formato json y tiene el siguiente aspecto:

```
{"topic_12": [["dalla", "0.051554136"], ["tx", "0.025218064"],
    ["amp", "0.013552489"], ["texa", "0.010390841"]], "topic_1
   4": [["tx", "0.09910201"], ["dalla", "0.08564694"], ["job",
    "0.07351037"], ["hire", "0.05538197"], ["great", "0.050553
   277"]], ...}
```

Listing 4.4: Ejemplo salida de LDA

4.2 Clasificación

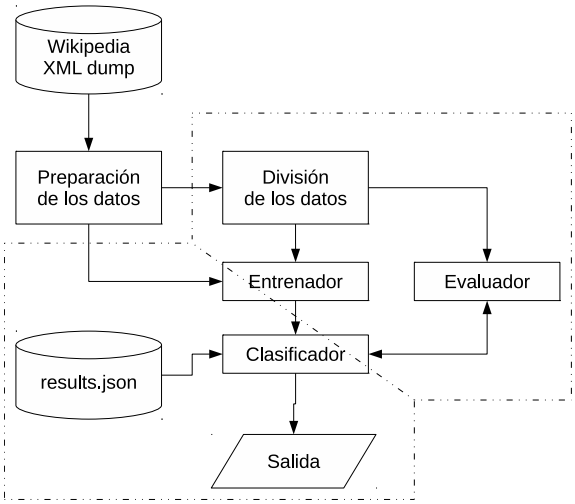

Figura 4.2: Diagrama implementación apartado clasificador

En la figura 4.2 se muestra con mayor detalle la parte del clasificador y evaluador vistas en la figura 3.1. Lo primero de todo es procesar el XML de la Wikipedia y crear un *dataset* con los artículos de las categorías que son relevantes. Se puede observar una división del esquema. Por una parte el entrenador y el clasificador son utilizados junto al evaluador para obtener una métrica del modelo. Por otra parte el entrenador recibe el conjunto completo de datos y clasifica la salida del bloque de LDA, imprimiendo por salida estándar los resultados de la asignación de categorías a los *topics*.

Para la realización del clasificador la estructura de datos de tipo diccionario es idónea ya que tenemos tuplas *clave-valor* en todas las fórmulas, como por ejemplo los pesos. Debido a que no es posible cargar en memoria principal todos los datos de los diccionarios, se ha optado por la librería *Shove*[3]. Dicha librería almacena los datos en disco y sólo tiene una parte en memoria principal, por lo que las restricciones de memoria RAM y la persistencia de los datos tras su cálculo se ve solucionado con una sola librería, eso sí, a cambio de una disminución del rendimiento al usarse memoria secundaria.

Para este apartado se han definido dos objetos llamados ***Article*** y ***Title***, ambos han requerido de una redefinición del comparador *eq* de Python. En la figura 4.3 se pueden observar los elementos que los componen.

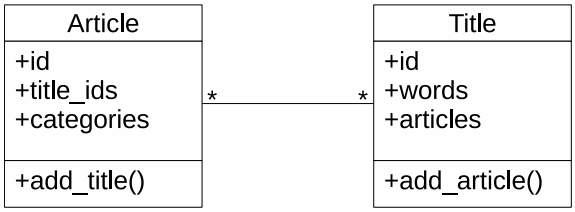

Figura 4.3: Diagrama clases *Article* y *Title*

4.2.1. Obtención de categorías

La obtención de las categorías pertenece a la preparación de los datos en la figura 4.2, ya que un paso importante es quedarnos con los artículos que tengan categorías que nos sean de interés, aquellos que tras la limpieza de categorías no dispongan de ninguna, serán eliminados del *dataset* final.

Las categorías de Wikipedia se encuentran en un archivo en formato SQL[4], por lo que para poder obtenerlas lo primero de todo ha sido necesario instalar MySQL en el equipo e importar el fichero con las categorías. La librería empleada para la conexión es PyMySQL[5]. Se conecta a la base de datos, obtiene todas las categorías que tengan entre *min* y *max* número de artículos y acto seguido hace una limpieza, eliminando aquellas que se hayan definido en un archivo de configuración indicando que no son relevantes.

Por considerarse irrelevantes ya porque no aportan un tema de diálogo en un tuit o bien porque son demasiado genéricas, se eliminan todas aquellas categorías que contangan:

actors, albums, articles, athletes, bases, boxers, by, champions, characters, cleanup, competitions, counties, cyclones, deaths, descent, episodes, equestrians, films, finals, footballers, from, images, in, members, musicians, needing confirmation, of, players, politicians, scientists, seasons, singers, songs, surnames, teams, throwers, winners, writers, AfC submissions, Articles, Articles that may contain, Articles to be expanded, Incomplete lists, Monthly clean-up category, People from, Portal-Class, Redirect-Class, Redirects, related

[3] `https://pypi.org/project/shove/`
[4] Structured Query Language. `https://www.w3schools.com/sql/sql_intro.asp`
[5] `https://github.com/PyMySQL/PyMySQL`

lists, requested images, stub, template, Unassessed, User, Wikipedia, WikiProject, Writers from.

Además se eliminan también aquellas categorías pertenecientes a años, décadas o que sólo son números. Además se cambian los guiones bajos por espacios ya que es en dicho formato como aparecen referenciadas en los artículos. Finalmente se devuelve en un conjunto al programa que llamó a la función.

4.2.2. Creación y limpieza del *dataset*

El conjunto de artículos de Wikipedia se encuentra en formato XML[6] donde se encuentran todas las páginas existentes, a continación se muestra un ejemplo reducido de una entrada.

```
<page>
    <title>AccessibleComputing</title>
    <ns>0</ns>
    <id>10</id>
    <redirect title="Computer accessibility" />
    <revision>
      <id>767284433</id>
      <parentid>631144794</parentid>
      <timestamp>2017-02-25T00:30:28Z</timestamp>
      <contributor>
        <username>Godsy</username>
        <id>23257138</id>
      </contributor>
      <comment>[[Template:This is a redirect]] has been deprecated, change
          to [[Template:Redirect category shell]].</comment>
      <model>wikitext</model>
      <format>text/x-wiki</format>
      <text xml:space="preserve">#REDIRECT [[Computer accessibility]]

{{Redirect category shell|
{{R from move}}
{{R from CamelCase}}
{{R unprintworthy}}
}}</text>
      <sha1>ds1crfrjsn7xv73djcs4e4aq9niwanx</sha1>
    </revision>
</page>
```

Listing 4.5: Ejemplo de página de la Wikipedia

Nos interesan sólo aquellas entradas cuyo *namespace* (*ns*) es cero ya que son los artículos propiamente dichos. En este caso se puede ver un ejemplo de redirección.

```
def process_articles(file_path):
    redirect_dict = {}
    file = open(file_path + "_temp", "w")
    pages = 0
    p_category = re.compile(r"\[\[Category:([^\]]*)\]\]")
    p_redirect = re.compile(r"#REDIRECT\ ?\[\[([^\]]*)\]\]")

    lines = []

    for event, elem in ET.iterparse(config_pp.get("articles_path"), events=(
        'start', 'end')):
        if event == "end" and elem.tag.endswith('page'):
            article_categories = []
```

[6] https://www.w3.org/XML/

```
            for item in list(elem):
                if item.tag.endswith('title'):
                    article_title = item.text
                elif item.tag.endswith('ns'):
                    article_ns = item.text
                    if article_ns != "0":
                        break
                elif item.tag.endswith('id'):
                    article_id = item.text
                elif item.tag.endswith('revision'):
                    for sub_item in list(item):
                        if sub_item.tag.endswith('text'):
                            article_text = sub_item.text

            if article_ns == "0":
                for match in re.finditer(p_category, article_text):
                    article_categories.append(match.group(1))

            if article_categories:
                lines.append(process_article(article_id, article_title,
                    article_text, article_categories))
                redirections = redirect_dict.pop(article_title, [])
                for redirection in redirections:
                    lines.append(process_article(article_id, redirection,
                        article_text, article_categories))

            elif "REDIRECT" in article_text:
                destinations = []
                for match in re.finditer(p_redirect, article_text):
                    destinations.append(match.group(1))
                for destination in destinations:
                    entry = redirect_dict.get(destination, [])
                    entry.append(article_title)
                    redirect_dict[destination] = entry

            elem.clear()
            pages += 1

            if pages %10000 == 0:
                file.writelines(lines)
                print("Processed pages = ", pages)
                lines = []
```

Listing 4.6: Procesado del XML para la obtención de los artículos

En esta primera parte se puede observar cómo se va iterando sobre el XML, como no cabe en memoria principal, se entra en el bucle a base de eventos, concretamente cuando se termina de leer un a página (que al finalizar la iteración es borrada de la memoria).

Cuando se tienen toda la información del artículo, se procesa para añadirlo al archivo del dataset. Nótese como se consulta si existen redirecciones para el artículo en cuestión, en caso de ser afirmativo, se añaden como si fuesen nuevos artículos pero en realidad mantienen el mismo id.

En caso de que se trate de una redirección, es añadida al diccionario en espera de encontrar el artículo al que apunta. Puede darse el caso en el que el artículo apuntado ya ha sido procesado, es por ello por lo que luego se realiza una segunda pasada sobre el dataset en busca de los artículos de destino para añadir las redirecciones.

```
    print("Total pages processed = ", pages)
    log.info("Closing temp file")
    file.writelines(lines)
    file.close()

```

```
    print("Writting dataset")
    output_file = open(file_path, "w")
    lines = []
    with open(file_path + "_temp", "r") as temp_file:
        for line in temp_file:
            lines.append(line)
            items = line[:-1].split(";")
            line_title = items[1]
            redirections = redirect_dict.pop(line_title, [])
            for redirection in redirections:
                lines.append(process_article(items[0], redirection, items
                    [2], items[3]))
            if len(lines) >= 100000:
                output_file.writelines(lines)
                lines = []

    output_file.writelines(lines)
    output_file.close()
    try:
        os.remove(file_path + "_temp")
    except OSError:
        pass
```

Listing 4.7: Inclusión de las redirecciones todavía no resueltas

Si bien para entrenar no nos interesa, para clasificar el documento sí que nos es necesaria la información del tf_w, es por ello que se procesa el artículo antes de ser escrito en el fichero de salida.

```
def process_article(id, title, text, categories):
    text_clean = clean_text_list(text)
    increment = 1 / len(text_clean)
    tf_dict = {}
    for term in text_clean:
        tf = tf_dict.get(term, 0)
        tf += increment
        tf_dict[term] = tf
    return str.join(";", [id, title, str([(x, tf_dict.get(x)) for x in
        tf_dict]), str(categories)]) + "\n"
```

Listing 4.8: Filtrado de categorías y obtención del tf_w

4.2.3. Aleatorización del *dataset*

Básicamente es el paso a código de la transformación Schwartziana, la ordenación y la unión en k-vías.

Se genera un listado con índices de 0 a el número de líneas del dataset y se aleatorizan, entonces a cada línea del dataset se le va asignando un índice del listado.

```
def add_indexes(file_path, file_path_temp):
    num_lines = sum(1 for line in open(file_path, "r"))
    indexes = list(range(num_lines))
    shuffle(indexes)
    lines = []
    index = 0
    output_file = open(file_path_temp + "_indexed", "w")
    with open(file_path, "r") as file:
        for line in file:
            lines.append(str(indexes[index]) + "#" + line)
            if len(lines) > 100000:
                output_file.writelines(lines)
```

```
                lines = []
                print("Wroted", str(index), "lines")
            index += 1
    output_file.writelines(lines)

    output_file.close()

    return num_lines
```

Listing 4.9: Indexación aleatoria del *dataset*

Para una mayor facilidad de manejo (y por falta de memoria) se divide en bloque el dataset indexado.

```
def divide_in_blocks(file_path_temp, block_size=100000, num_lines=0):
    if num_lines == 0:
        num_lines = sum(1 for line in open(file_path_temp + "_indexed", "r")
            )

    num_blocks = int(math.ceil(num_lines / block_size))

    index = 0
    dataset_file = open(file_path_temp + "_indexed", "r")
    for block in range(num_blocks):
        print(str.format("Writting block number {}", str(block)))
        block_file = open(file_path_temp + "_block_" + str(block), "w")
        lines = []
        for x in range(block_size):
            if index < num_lines:
                lines.append(dataset_file.readline())
                index += 1
        block_file.writelines(lines)
        block_file.close()

    dataset_file.close()

    return num_blocks
```

Listing 4.10: División en bloques del *dataset* indexado

Se ha ordenado cada bloque empleando el algoritmo de ordenación que implementa Python, ha sido necesario redefinir el comparador para que ordene en función del índice.

```
def sort_blocks(file_path_temp, num_blocks):
    for block in range(num_blocks):
        print("Sorting block", str(block))
        sort_block(file_path_temp + "_block_" + str(block))


def sort_block(file_path):
    file = open(file_path, "r")
    lines = file.readlines()
    file.close()
    lines.sort(key=cmp_to_key(compare_lines))
    file = open(file_path, "w")
    file.writelines(lines)
    file.close()


def compare_lines(line_a, line_b):
    index_a = int(line_a.split("#")[0])
    index_b = int(line_b.split("#")[0])
    comparation = index_a - index_b
    if comparation < 0:
```

```
        result= -1
    elif comparation > 0:
        result = 1
    else:
        result = 0
    return result
```

Listing 4.11: Ordenación de los bloques

Finalmente el algoritmo de unión, es necesario definir funciones para trabajar con buffers de los bloques de forma dinámica.

```
def k_way_merge(file_path_output, file_path_temp, num_blocks, buffer=100000):
    files_dict = {}
    files_lines_dict = {}
    output_buffer = []
    output_file = open(file_path_output, "w")
    for block in range(num_blocks):
        block_file = open(file_path_temp + "_block_" + str(block), "r")
        files_dict[block] = block_file
        files_lines_dict[block] = block_file.readlines(100)

    done = False
    while not done:
        min_block = -1
        min_line = ""
        keys = files_lines_dict.keys()
        for key in keys:
            line = get_line(files_dict, files_lines_dict, key)
            if len(line):
                if min_block < 0 or compare_lines(line, min_line) < 0:
                    min_line = line
                    min_block = key
        if len(min_line):
            output_buffer.append(min_line.split("#")[1])
            pop_line(files_dict, files_lines_dict, min_block)
            if len(output_buffer) >= buffer:
                output_file.writelines(output_buffer)
                output_buffer = []
                print(str.format("Wtritten {} lines into {}", buffer,
                    file_path_output))
        else:
            done = True

    output_file.writelines(output_buffer)
    output_file.close()


def get_line(files_dict, files_lines_dict, block):
    result = ""
    if block in files_lines_dict:
        lines = files_lines_dict.get(block, [])
        if len(lines):
            result = lines[0]
        else:
            file = files_dict.get(block)
            lines = file.readlines(100)
            files_lines_dict[block] = lines
            if len(lines):
                result = lines[0]
    return result


```

```
51 def pop_line(files_dict, files_lines_dict, block):
52     line = get_line(files_dict, files_lines_dict, block)
53     if len(line):
54         files_lines_dict[block] = files_lines_dict[block][1:]
```

Listing 4.12: Unión en k-vías

4.2.4. Partición de datos

A la hora de particionar los datos para la validación cruzada, se indica en cuántos bloques se va a dividir el *dataset*, calcula el tamaño de bloque y tras pasarle qué bloque va a ser el de test, recorre todo el fichero fuente llevando un índice de qué línea se trata y dependiendo de este y los límites del bloque de test, se escribe en *train* o *test*. Por motivos de eficiencia, se guardan las líneas en un *buffer* y al llegar al máximo es cuando se accede al disco duro para escribir los datos. Cuando ya no queden más lineas que leer escribe el contenido de los *buffers* y cierra los archivos.

```
1     first_test = block_size * block
2     last_test = min((block + 1) * block_size, num_lines)
3
4     test_file = open(output_path + "test_" + str(block), "w")
5     train_file = open(output_path + "train_" + str(block), "w")
6     lines_test = []
7     lines_train = []
8
9     max_items = 100000
10    index = 0
11
12    with open(input_path, "r") as file:
13        for line in file:
14            if first_test <= index < last_test:
15                lines_test.append(line)
16            else:
17                lines_train.append(line)
18
19            if len(lines_test) > max_items:
20                test_file.writelines(lines_test)
21                lines_test = []
22            if len(lines_train) > max_items:
23                train_file.writelines(lines_train)
24                lines_train = []
25            index += 1
26
27    test_file.writelines(lines_test)
28    train_file.writelines(lines_train)
29    test_file.close()
30    train_file.close()
```

Listing 4.13: División de los datos en **entrenamiento** y **test**

4.2.5. Entrenador

En este apartado se procesan todos los artículos de Wikipedia que se encuentran en el *dataset* y se construyen los diccionarios que contienen la información que se mostraba en 3.3.

Lo primero de todo es la creación de los diccionarios, algunos solo son necesarios en este apartado y el volumen de datos cabe en memoria principal, por lo que ni serán persistidos ni se requiere de *Shove* para su utilización.

```
articles_dict = Shove("file://" + shove_folder + "/articles", sync=
    shove_buffer)
titles_dict = Shove("file://" + shove_folder + "/titles", sync=shove_buffer)
words_dict = Shove("file://" + shove_folder + "/words", sync=shove_buffer)
categories_dict = {}
cf_w_dict = {}
R_w_dict = Shove("file://" + shove_folder + "/R_w", sync=shove_buffer)
vocabulary_c_dict = Shove("file://" + shove_folder + "/vocabulary_c", sync=
    shove_buffer)

p_nonword = re.compile(r'\W')
```

Listing 4.14: Creación de diccionarios, varios de ellos con *Shove*

Procesado del *dataset*

Se recorre línea a línea del *dataset* de entrenamiento, extrayendo la información necesaria de cada artículo y luego se llama a la función encargada de procesarlo.

```
def process_articles():
    pages = 0
    with open(dataset, "r") as data:
        for line in data:
            elements = line.split(";")
            article_id = elements[0]
            article_title = elements[1]
            categories_part = elements[3][1:-2]
            categories_part = categories_part.replace("\'", "")
            article_categories = [category.strip() for category in
                categories_part.split(",")]

            if article_categories:
                article = Article(id=article_id, categories=
                    article_categories)
                process_article(article, article_title)
                del article

            pages += 1
            if pages %10000 == 0:
                print("Processed pages = ", pages)
```

Listing 4.15: Procesado de los artículos de **entrenamiento**

Procesado del artículo

El procesado del artículo primero requiere del procesado del título, ya que si este no puede ser creado (no devuelve un objeto del tipo ***Title***) no tiene sentido procesar el artículo.

```
def process_article(article, title):
    title_entry = process_title(title, article)
    if type(title_entry) == Title:
        for category in article.categories:
            add_article_to_category(article, category)
            for word in title_entry.words:
                add_w_to_vocabulary_c(word, category)

        if title_entry.id not in set(article.title_ids):
            article.add_title(title_entry.id)
        if article.id not in articles_dict or article != articles_dict.get(
            article.id):
```

```
12              articles_dict[article.id] = article
13      del article
```

Listing 4.16: Procesado del artículo extraído del *dataset* de **entrenamiento**

Procesado del título

Debido a que el título es un conjunto de palabras, este no puede ser la clave de un diccionario, es por ello que se utiliza la función *frozenset* que crea un objeto del mismo tipo, siendo un conjunto invariable y que permite ser utilizado como índice.

Sin embargo *frozenset* no es suficiente, como se va a persistir en disco con *Shove* y la clave de la entrada en el diccionario es el nombre del archivo en el sistema, hay que convertir todo símbolo no permitido a "_" y a su vez hay que realizar una comprobación de que el nombre del fichero (tras la conversión de ciertos caracteres unicode) no excede el límite máximo del sistema operativo (en este caso GNU/Linux).

Si el título es válido se almacena la estructura de la información.

```
1  def process_title(title, article):
2      title_set = clean_text(title)
3
4      id = frozenset_to_filename(frozenset(title_set))
5      if len(id) >= 254 or len(pathname2url(id)) >= 254:
6          return "error"
7      else:
8          try:
9              title_entry = titles_dict.get(id, Title(id=id, words=title_set))
10             if article.id not in set(title_entry.articles):
11                 title_entry.add_article(article.id)
12                 titles_dict[id] = title_entry
13
14             add_title_to_words(title_entry)
15             return title_entry
16         except:
17             return "error"
```

Listing 4.17: Procesado del título

Precálculo de cf_w

Con el fin de agilizar luego la clasificación, durante la fase del entrenamiento se precalcula la parte de cf_w que no depende del peso de la palabra en el documento, luego en la clasificación solo tendrá que ser accedida la entrada en el diccionario y multiplicar su peso por el valor ya calculado. Esta parte es muy costosa por el tema del acceso a disco, es por ello por lo que es preferible precalcularla ahora.

```
1  def calculate_cf_w():
2      for word in words_dict:
3          cf_w = set()
4          word_entry = words_dict.get(word)
5          for title in word_entry:
6              title_entry = titles_dict.get(title)
7              for article in title_entry.articles:
8                  article_entry = articles_dict.get(article)
9                  cf_w = cf_w.union(article_entry.categories)
10         cf_w_dict[word] = len(cf_w)
11         R_w_dict[word] = math.log(len(categories_dict) / len(cf_w), 2)
```

Listing 4.18: Precálculo de cf_w

Otras funciones de ayuda

```
def add_title_to_words(title):
    for word in title.words:
        word_entry = words_dict.get(word, [])
        if title.id not in set(word_entry):
            word_entry.append(title.id)
            words_dict[word] = word_entry
```

Listing 4.19: Vínculo título-palabras

```
def add_article_to_category(article, category):
    articles = categories_dict.get(category, [])
    if article.id not in set(articles):
        articles.append(article.id)
        categories_dict[category] = articles
```

Listing 4.20: Vínculo entre artículos y categorías

```
def frozenset_to_filename(x):
    return re.sub(p_nonword, "_", str(x))
```

Listing 4.21: Conversión de *frozenset* a un string que pueda ser nombre de fichero

```
def add_w_to_vocabulary_c(word, category):
    words_c = vocabulary_c_dict.get(category, set())
    if word not in words_c:
        words_c.add(word)
        vocabulary_c_dict[category] = words_c
```

Listing 4.22: Inclusión de w en el vocabulario de c

4.2.6. Clasificador

El clasificador lo primero de todo que hace es acceder a los diccionarios almacenados mediante *Shove* en el entrenamiento. En esta fase espera como entrada un documento compuesto por una lista de tuplas compuestas por la palabra y su peso asociado tras el paso por LDA (o su *tf* si proviene del evaluador). En esta sección se realizan todas las fórmulas vistas en el modelado, sección 3.4.2.

Lo primero es la carga de los diccionarios, los objetivos de crear un conjunto con las claves del diccionario de palabras son dos, 1) no tener que consultar *Shove* constantemente lo que ralentizaría el proceso, y 2) consultar si una palabra pertenece a un conjunto es más eficiente que consultar si pertenece a una lista.

```
articles_dict = Shove("file://" + shove_folder + "/articles", sync=
    shove_buffer)
titles_dict = Shove("file://" + shove_folder + "/titles", sync=shove_buffer)
words_dict = Shove("file://" + shove_folder + "/words", sync=shove_buffer)
R_w_dict = Shove("file://" + shove_folder + "/R_w", sync=shove_buffer)
vocabulary_c_dict = Shove("file://" + shove_folder + "/vocabulary_c", sync=
    shove_buffer)

words_set = set(words_dict)
```

Listing 4.23: Acceso a los diccionarios con los datos

Cuerpo principal

Esta función va llamando a las funciones que implementan las fórmulas vistas en la sección 3.4.2. La función **prepare_words** procesa la lista de tuplas en algo que se pueda manejar de forma sencilla así como filtrar las palabras dejando aquellas que aparecen en el sistema entrenado.

```
def clasify_topic(topic, topic_id, num_results=10):
    log.info("[%s] Filtering words", topic_id)
    words_topic, words_weight_dict = prepare_words(topic)
    log.info("[%s] w->t", topic_id)
    w_supports_t = calculate_w_supports_t_and_S_t(words_topic)
    log.info("[%s] B_c", topic_id)
    B_c_dict = calculate_B_c(w_supports_t)
    log.info("[%s] R_t", topic_id)
    R_t_dict = calculate_R_t(w_supports_t, words_weight_dict)
    log.info("[%s] R_a", topic_id)
    R_a_dict = calculate_R_a(R_t_dict)
    log.info("[%s] R_c", topic_id)
    R_c_dict = calculate_R_c(R_a_dict, B_c_dict)

    log.info("[%s] R_c_prime", topic_id)
    R_c_list = sorted(list(R_c_dict.items()), key=lambda x: x[1], reverse=
        True)
    d_w_dict = create_d_w(words_set)
    R_c_prime_list = []
    for R_c in R_c_list:
        R_c_prime_list.append(recalculate_R_c(R_c, B_c_dict.get(R_c[0]),
            d_w_dict))

    result = sorted(R_c_prime_list, key=lambda x: x[1], reverse=True)[0:
        num_results]
    print("[", topic_id, "] Classification: ", result)
    log.info("[%s] Words unfiltered: %s", topic_id, topic)
    log.info("[%s] Words filtered (%d): %s", topic_id, len(words_topic),
        words_topic)
    return result
```

Listing 4.24: Clasificación del documento

```
def prepare_words(topic):
    words_tuples = [word for word in topic if word[0] in words_set]
    words_weight_dict = {}
    words_topic = []
    for tuple in words_tuples:
        word = tuple[0]
        weight = float(tuple[1])
        words_weight_dict[word] = weight
        words_topic.append(word)
    return words_topic, words_weight_dict
```

Listing 4.25: Conversión de tuplas al formato adecuado

Cálculo de B_c

Para obtener dicho cálculo primero se necesita conocer las *supporting words* de los títulos para acto seguido poder calcular las *supporting words* de cada categoría c

```
def calculate_w_supports_t_and_S_t(words_topic):
    words_topic_set = set(words_topic)
    w_supports_t = {}
    for word in words_topic:
```

```
        word_entry = words_dict.get(word)
        for title_id in word_entry:
            title_entry = titles_dict.get(title_id)
            if len(title_entry.words) == 1:
                w_supports_t[(word, title_id)] = 1
            else:
                S_t = len(title_entry.words.intersection(words_topic_set))
                if S_t >= len(title_entry.words) - 1:
                    w_supports_t[(word, title_id)] = S_t
    return w_supports_t
```

Listing 4.26: Cálculo de las *support words* de t

```
def calculate_B_c(w_supports_t):
    B_c_dict = {}
    for w_t in w_supports_t:
        word = w_t[0]
        title_id = w_t[1]
        title_entry = titles_dict.get(title_id)
        for article in title_entry.articles:
            article_entry = articles_dict.get(article)
            for category in article_entry.categories:
                B_c = B_c_dict.get(category, set())
                B_c.add(word)
                B_c_dict[category] = B_c
    return B_c_dict
```

Listing 4.27: Cálculo de B_c

Cálculo de R_t

Se calculan los pesos de aquellos títulos que son *soportados* por el documento a clasificar. obsérvese cómo R_w que había sido calculado parcialmente en el entrenamiento ahora está siendo multiplicado por el peso de la palabra.

```
def calculate_R_t(w_supports_t, words_weight_dict):
    R_t_dict = {}
    for w_t in w_supports_t:
        word = w_t[0]
        title_id = w_t[1]
        title_entry = titles_dict.get(title_id)
        w_w = words_weight_dict.get(word, 1)
        R_w = R_w_dict.get(word)
        t_w = len(words_dict.get(word))
        a_t = len(title_entry.articles)
        S_t = w_supports_t.get(w_t)
        L_t = len(title_entry.words)
        R_t = R_t_dict.get(title_id, 0)

        value = R_t + (w_w * R_w * (1 / t_w) * (1 / a_t) * (S_t / L_t))
        R_t_dict[title_id] = value
    return R_t_dict
```

Listing 4.28: Cálculo del peso de los títulos

Cálculo de R_a

```
def calculate_R_a(R_t_dict):
    R_a_dict = {}
    for title_id in R_t_dict:
        R_t = R_t_dict.get(title_id)
        title_entry = titles_dict.get(title_id)
        for article in title_entry.articles:
            R_a = R_a_dict.get(article, 0)
            if R_t > R_a:
                R_a = R_t
            R_a_dict[article] = R_a
    return R_a_dict
```

Listing 4.29: Cálculo del peso de los artículos

Cálculo de R_c

Finalmente tenemos el cálculo del peso de las categorías.

```
def calculate_R_c(R_a_dict, B_c_dict):
    R_c_dict = {}
    for article in R_a_dict:
        article_entry = articles_dict.get(article)
        R_a = R_a_dict.get(article)
        for category in article_entry.categories:
            R_c = R_c_dict.get(category, 0)
            R_c_dict[category] = R_c + R_a
    for category in R_c_dict:
        R_c = R_c_dict.get(category, [])
        v_c = len(B_c_dict.get(category, []))
        d_c = len(vocabulary_c_dict.get(category, []))
        if d_c == 0:
            R_c_dict[category] = 0
        else:
            R_c_dict[category] = (v_c / d_c) * R_c
    return R_c_dict
```

Listing 4.30: Cálculo del peso de las categorías

Cálculo de R'_c

Para este proceso primero se establece el valor de *decay* de cada palabra en 1 y se recorren las categorías de mayor a menor peso, modificando su valor en función de las fórmulas vistas en el modelado de la sección 3.4.2.

```
def create_d_w(words):
    d_w_dict = {}
    for word in words:
        d_w_dict[word] = 1
    return d_w_dict
```

Listing 4.31: Inicialización del *decay*

```
def recalculate_R_c(R_c, B_c, d_w_dict):
    sumatory = 0
    for word in B_c:
        d_w = d_w_dict.get(word)
        sumatory += d_w
        d_w_dict[word] = d_w / 2
    return (R_c[0], R_c[1] * (float(sumatory) / float(len(B_c))))
```

Listing 4.32: Ajuste de R_c en función del *decay*

4.2.7. Evaluador

Para la evaluación del sistema, se ha creado un archivo Python que va leyendo los documentos en el conjunto de **test**, extrae las categorías y le va pasando al clasificador los documentos como si fuesen la salida de LDA. Se hace un sumatorio de los falsos positivos, falsos negativos y verdaderos positivos, tanto a nivel de documento como agregando a nivel global.

```
def test_system(test_dataset):
    index = 0
    true_positives_global = 0
    false_positives_global = 0
    false_negatives_global = 0
    file_results = open("./test_results", "a")
    with open(test_dataset, "r") as file:
        for line in file:
            print("Calculating line:", index)
            true_positives = 0
            false_positives = 0
            false_negatives = 0
            f1_score = 0
            index += 1
            elements = line.split(";")
            article_id = elements[0]
            words_part = elements[2][2:-2]
            words_part = words_part.replace("\'", "").split("), (")
            article_words = []
            for entry in words_part:
                splitted = entry.split(",")
                article_words.append((splitted[0].strip(), splitted[1].strip
                    ()))
            categories_part = elements[3][1:-2]
            categories_part = categories_part.replace("\'", "")
            article_categories = [category.strip() for category in
                categories_part.split(",")]

            result = clasify_topic(article_words, article_id, num_results=
                len(article_categories))
            result = [tuple[0] for tuple in result]
            for category in result:
                if category in article_categories:
                    true_positives += 1
                else:
                    false_positives += 1
            false_negatives += sum([1 for category in article_categories if
                category not in result])
            true_positives_global += true_positives
            false_negatives_global += false_negatives
            false_positives_global += false_positives
```

Listing 4.33: Evaluador, toma de datos

Se calculan las métricas de **precisión**, **exhaustividad** y **puntuación F1**, tanto a nivel de documento como global. Las del documento son guardadas en un archivo CSV junto a otros datos con la finalidad de un análisis futuro.

```
            if (true_positives + false_positives) > 0:
                precision = true_positives / (true_positives +
                    false_positives)
            if (true_positives + false_negatives) > 0:
                recall = true_positives / (true_positives + false_negatives)
            if (precision + recall) > 0:
                f1_score = 2 * (recall * precision) / (recall + precision)
            print(article_categories)
            print(str.format("Categories: {} | Precision: {} | Recall: {} |
                F1 Score: {}", len(article_categories),
                              precision, recall, f1_score))
            file_results.write(str.format("{}; {}; {}; {}; {}\n", article_id
                , len(article_categories), precision, recall, f1_score))

    precision = true_positives / (true_positives + false_positives)
    recall = true_positives / (true_positives + false_negatives)
    f1_score = 2 * (recall * precision) / (recall + precision)
    print(str.format("Precision: {} | Recall: {} | F1 Score: {}", precision,
        recall, f1_score))
```

Listing 4.34: Evaluador, cálculo de métricas

4.3 Problemas encontrados

A continuación se comentan algunos de los problemas acaecidos durante la realización del presente proyecto.

4.3.1. WordNet

Antes de realizar la clasificación de temas utilizando la Wikipedia, se probó con WordNet, aquí los principales problemas han sido:

- WordNet sólo asigna una categoría por significado, si una palabra pudiese pertenecer a dos dominios sólo aparece uno.
- La versión que se instala utilizando NLTK[7] en Python está incompleta, concretamente carece de los dominios.
- Existe un proyecto para poder mapear significados de palabras a dominios, llamado WNDomains [38], sin embargo este proyecto es para una versión previa de Wordnet, por lo que no era de utilidad.
- Existen otros proyectos abiertos como WNDomains pero ninguno cumplía características deseadas para el proyecto, algunos no trataban topics mientras que otros más que dominios de conocimiento eran casi sinónimos alternativos.

[7] *Natural Language Toolkit*, conjunto de herramientas para trabajar con lenguaje natural en Python. Herramientas tales como para la eliminación de las *stopwords*, reducción de las palabras a la raíz, obtención de información semántica, descarga de corpora, etc ... Más información en: `https://www.nltk.org/`

4.3.2. Ejecución

Durante el proceso de desarrollo y ejecución del proyecto estos han sido los principales problemas:

- Se ha configurado *Shove* para que escribiese en disco, esto ocasiona que:
  - Cada entrada en el diccionario es un archivo, siendo su clave el nombre, presenta problemas con la longitud máxima de fichero.
  - Se generan muchos archivos de poco tamaño, a pesar de existir espacio suficiente el sistema se queda sin *inodes* [8] disponibles y da error de que no hay espacio disponible.
  - la generación de muchos archivos pequeños ralentiza el sistema mucho más que escribir pocos de gran tamaño, esto se debe a que el número de accesos al disco para escribir aumenta considerablemente.
- Se intentó utilizar MySql con *Shove* para solventar los problemas relacionados con escribir directamente en disco, pero el tiempo de procesado aumentaba de forma considerable (lo que antes podía tardar unos minutos pasaba a tardar horas).
- Previo al uso de Wikipedia se empleó en las pruebas Wiktionary, que tiene un volumen de datos mucho menor. A pesar de que ambos proyectos siguen la misma base, la definición de las categorías y su indicación en los artículos no es la misma, por lo que parte del trabajo realizado y probado con Wiktionary no servía para Wikipedia.
- Debido a las limitaciones del equipo, hacer una validación cruzada con una distribución 90 %-10 % resulta imposible por el tiempo requerido para evaluar un solo par **entrenamiento-test**
- El delimitador empleado en origen para la transformación Schwartziana entraba en conflicto con el propio texto del artículo. Se utilizaba el símbolo % y este se emplea para la codificación de ciertos símbolos especiales, esto ocasionaba una serie de errores durante el entrenamiento y clasificación.

4.3.3. Soluciones

Algunas de las medidas tomadas para resolver las incidencias anteriormente citadas, han sido:

- Detectar los artículos cuyo nombre de fichero sea más largo que el permitido por el sistema de archivos de Linux y evitar procesarlos.
- Para acelerar los tiempos de ejecución se ha distribuido todo lo posible en distintos discos duros para que se pudiese leer y escribir en distintos sitios a la vez, y finalmente se añadió también un SSD.
- Se descartó por completo usar MySql y se continuó con el sistema de escritura en disco entrada-archivo.
- Rehacer las partes no compatibles de Wiktionary con Wikipedia y aplicar un mayor énfasis en la limpieza de categorías irrelevantes.

[8] Un *inode* es una estructura de datos propia de los sistemas de archivos tradicionalmente empleados en los sistemas operativos tipo UNIX como es el caso de Linux. Un inodo contiene las características de un archivo regular, directorio, o cualquier otro objeto que pueda contener el sistema de ficheros.[39]

CAPÍTULO 5
Resultados obtenidos

En este apartado se verá la evaluación del sistema entrenado a través de la evaluación cruzada vista en el modelado (sección 3.5), así como aplicado a la salida del algoritmo LDA procesando tuits de varias ciudades de EE.UU.

5.1 Clasificación artículos de Wikipedia (validación cruzada)

Los resultados que obtuvo Shönhofen [30] son de un mínimo de 0.2 y un máximo de 0.5 en la precisión (*precision*) y una media del 0.5 en la exhaustividad (*recall*), se tomarán estos valores como referencia comparativa.

Debido a las limitaciones del equipo donde se han realizado las pruebas, ha resultado totalmente inviable realizar una validación cruzada tal y como se definió en el modelado (ver sección 3.5), es por ello que sólo se han podido realizar un número reducido de pruebas y con un conjunto de datos de test en cada par entrenamiento-test inferior al 1 % del total del *dataset* (poco más de diez mil artículos en el conjunto de test).

5.1.1. Sistema entrenado con un millón de artículos

Debido al tiempo que conlleva entrenar el sistema con el conjunto de entrenamiento en su totalidad, se realizó una primera prueba donde sólo se utilizó un millón de artículos para entrenar el sistema (en lugar de los más de ocho que contiene el conjunto de entrenamiento).

Tal y como puede verse en la figura 5.1, donde el *score* representa el grado de acierto en cada una de las métricas siendo 1 el mejor valor obtenible (100 % de acierto), existen disparidad de resultados, en algunos casos perfectos y en otros casos no se ha acertado nada. La gráfica muestra las métricas en forma de nube de puntos como en curvas aproximadas (meduiante la función *smooth scsplines* de *gnulot*[1] a los puntos.

[1] `http://www.gnuplot.info/`

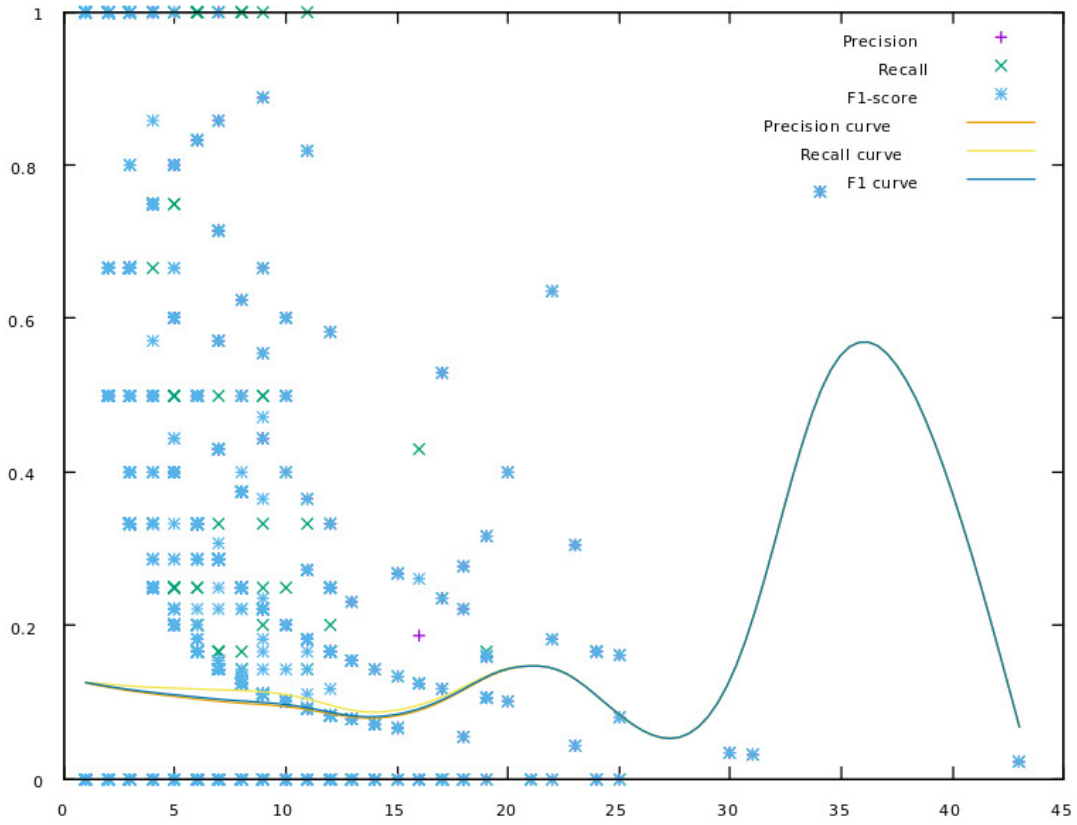

Figura 5.1: Gráfico resultados con entrenamiento reducido. Eje X número de categorías por artículo, eje Y valor de la métrica.

Los artículos con muchas categorías son menos frecuentes (tal y como puede verse en la tabla 5.1) y es por ello que la parte derecha de la gráfica contiene menor volumen de puntos y las curvas aproximadas a los datos son más vulnerables a los datos anormales en dicha zona.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	>15
4671	2559	1446	799	430	297	168	109	70	43	34	35	12	9	7	52

Tabla 5.1: Distribución de artículos de test por número de categorías

Si se realiza un análisis de las métricas a nivel global de forma agregada, se obtienen unos resultados inferiores a los del trabajo de Shönhofen tal y como se muestra en la tabla 5.2.

Precision	*Recall*	*F1-score*
0.112118	0.113267	0.112689

Tabla 5.2: Resultados agregados para entrenamiento reducido

5.1.2. Sistema con el dataset de entrenamiento completo

Al incrementar la cantidad de información contenida en el clasificador de un millón de artículos a más de ocho, la clasificación de test ha visto mermada su velocidad de cómputo, es por ello que también se reducen los artículos a clasificar de diez mil a mil.

Par entrenamiento-test 1

Se puede observar en la figura 5.2 cómo, a diferencia del sistema entrenado con un millón de artículos, se han mejorado los resultados de la clasificación, y en los resultados agregados de este par en la tabla 5.4 se muestra cómo el valor ha mejorado con respecto al corpus de entrenamiento reducido (tabla 5.2. Al igual que antes, la distribución de artículos por número de categorías es mayor cuantas menores son el numero de estas (tabla 5.3).

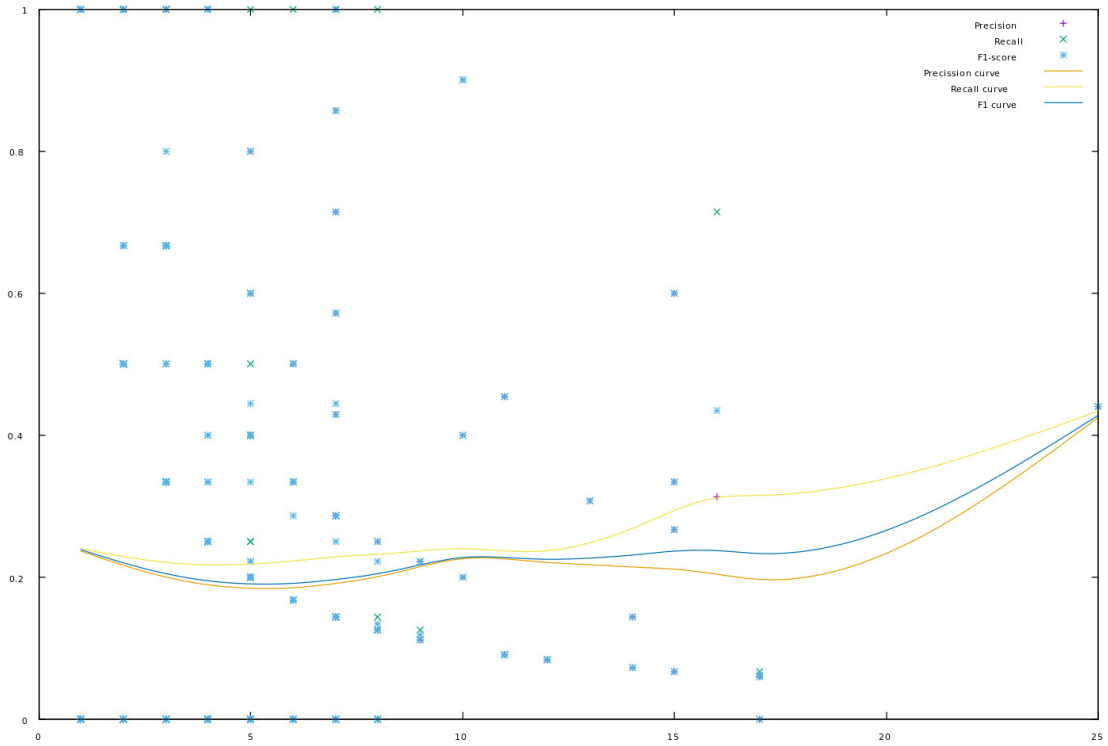

Figura 5.2: Gráfico resultados *cross validation* par 1. Eje X número de categorías por artículo, eje Y valor de la métrica.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	>15
473	260	143	71	45	25	21	12	4	3	4	2	1	2	4	5

Tabla 5.3: Distribución de artículos de test por número de categorías, par 1

Precision	*Recall*	*F1-score*
0.227835	0.239281	0.230652

Tabla 5.4: Resultados agregados par 1

Par entrenamiento-test 2

Si comparamos las gráficas 5.2 y 5.3 veremos que donde se encuentra el grueso de los artículos los resultados son similares, siendo en este caso también unos resultados superiores al 20 % (véase tabla 5.6

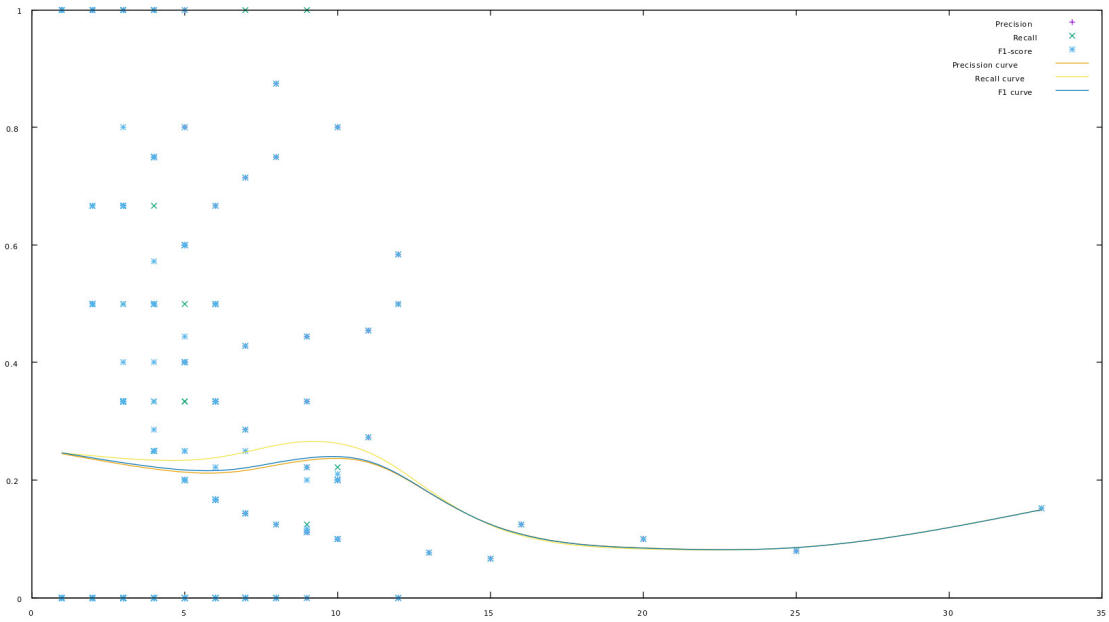

Figura 5.3: Gráfico resultados *cross validation* par 2. Eje X número de categorías por artículo, eje Y valor de la métrica.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	>15
464	241	148	90	49	36	13	6	8	8	2	4	1	0	1	4

Tabla 5.5: Distribución de artículos de test por número de categorías, par 2

Precision	*Recall*	*F1-score*
0.222717	0.226159	0.224425

Tabla 5.6: Resultados agregados par 2

5.1.3. Disparidad de resultados

La diferencia entre el trabajo de Shönhofen [30] y este puede deberse a discrepancias en la limpieza de categorías, al conjunto de datos utilizado o a la implementación realizada en cuanto al procesado de los artículos.

También hay que tener en cuenta que el objetivo principal de este trabajo es implementar una herramienta que sirva de ayuda para discernir de qué está hablando un tema de Twitter y, examinando los resultados obtenidos, parece ser que cumple el objetivo tal y como se puede ver en el siguiente artículo:

- ID artículo: 12475249

- Categorías asignadas:
  - Ukrainian classical violinists
  - Dora asteroids
  - Henryk Wieniawski Violin Competition prize-winners
  - French-Canadian families
- Categorías reales:
  - Soviet emigrants to Canada
  - Université de Montréal faculty
  - Soviet violinists
  - Male violinists

El resultado de la clasificación ha sido de 0 en cualquier métrica, pero si analizamos las categorías asignadas se puede deducir que se está hablando sobre un violinista, posiblemente emigrante, cosa que así es si miramos las categorías reales. Si se busca el artículo al que pertenece el identificador se descubre que se trata de *Vladimir Landsman*[2], un violinista nacido en la URSS, que ganó premios y emigró a Canadá donde obtuvo la nacionalidad. También se observa que el artículo es de una longitud reducida, lo que puede haber favorecido el fallo en la clasificación

Otro artículo de métrica cero es el siguiente:

- ID artículo: 3395369
- Categorías asignadas:
  - Northern & Shell
  - Syrian defectors
- Categorías reales:
  - Lebanese anti-Syrian activists
  - Lebanese socialites

Este artículo es una redirección de *Gebran Tueni*[3], concretamente el título del artículo que redirecciona hace referencia al que se creó tras la muerte del periodista, muy crítico con el gobierno Sirioy defensor de la libertad de prensa, en un atentado por coche bomba. Es por todo esto por lo que las categorías asignadas tienen relación con el periodismo y con Siria.

Veamos a continuación un artículo más con métrica cero:

- ID artículo: 24361255
- Categorías asignadas:
  - Komnenos dynasty
- Categorías reales:

[2] `https://en.wikipedia.org/wiki/Vladimir_Landsman`
[3] `https://en.wikipedia.org/wiki/Gebran_Tueni`

- Porphyrogennetoi

El artículo es una redirección de título *Theodora Komnene Angelos*[4]. Tras el procesado del dataset, el test ha debido de perder la categoría *Komnenos dynasty* ya que en la Wikipedia sí que aparece. Cabe destacar que aunque la evaluación indique que no es correcto, en realidad sí que lo es ya que la categoría asignada es una categoría real del artículo.

También puede deberse esta disparidad a que los resultados mostrados en Shönhofen [30] parecen estar generalizados, en la tabla de categorías asignadas que muestra, estas son mucho más genéricas que con las que se están utilizando en este proyecto, es posible que haya aplicado una generalización uniendo las distintas subcategorías como una única categoría padre.

5.2 Clasificación temas LDA

En esta sección se realizará una clasificación de los temas que se obtienen mediante LDA aplicado a los tuits de varias ciudades de EE.UU. recogidos durante el periodo electoral de 2016. Se compararán las categorías que se asignan mediante el sistema entrenado con el corpus reducido de un millón de artículos y el corpus completo con todos los artículos de Wikipedia.

El *dataset* con los tuits es el mismo que el utilizado en [11], por lo que se utiliza para cada ciudad el mismo número de temas que se utilizó en dicho trabajo tal y como se muestra en la tabla 5.7.

Ciudad	Número de temas
Chicago	15
Dallas	16
Denver	14
Las Vegas	15
Los Ángeles	13
Nueva York	19
Phoenix	17
San Francisco	13
Washington	18

Tabla 5.7: Número de temas por ciudades

A continuación se muestran las tablas con las categorías asignadas a los cinco primeros *topics* de cada ciudad (los resultados completos pueden verse en el anexo A.2), clasificando la salida de LDA con las diez primeras palabras obtenidas por dicho algoritmo. El listado de categorías asignadas está ordenado por orden de relevancia según el clasificador, por lo que en el listado el primer resultado tendrá mayor peso que el último (los valores numéricos se han omitido por tema de formato de las tablas, los valores completos se encuentran en el repositorio del proyecto[5]).

[4] `https://en.wikipedia.org/wiki/Theodora_Komnene_Angelos`
[5] `https://github.com/carlos3dx/tfm2018`

5.2.1. Corpus reducido

Topic	Palabras	Categorías
0	map, legoland, hotchocolaterun, themadnesstour, themakeupshow, religi, southwest, bae, ceremoni, uicrha	Legoland, Eritrean culture, Royal Saudi Air Force
1	ctuari, southeast, shedd, greet, gas, internet, lash, fix, liberti, sight	Dutch female rowers, Insurance, Fax software
2	reject, nellcôt, corridor, hairstylist, rue, piri, skydeckchicago,lovefashionletstalk, immedi, dbzkai	French Riviera, Thoroughbred family 2-e, Sapindales families
3	respect, renaiss, trick, velvet, thai, alien, concept, copywrit, bts,puppi	Advertising occupations, Non-fiction books about acting, Journalism occupations
4	campus, best, midwestlegend, yellowcard, scorpio, wermus, portag,themakeupshowchicago, behavior, backstagel	Nick Fury, Kursk submarine disaster, Robotic submarines

Tabla 5.8: Asignaciones a los *topics* para Chicago (corpus reducido)

Topic	Palabras	Categorías
0	walnut, crab, growth, chapel, hrh, liveroyali, riverfront, boil, iron, ojo	Surat, Hot springs, Kolkata, Anomura
1	pari, prayforpari, franc, terror, rapper, boosi, swim, nomad, fenomenal, soire	Napier aircraft engines, McKinley Senior High School alumni, Flat engines
2	gogolbordello, momo, princess, np, gypsypunk, clip, doughnut, iren, semifin, trainer	Brazilian Carnival, United States intelligence operations, Tournament systems
3	parad, brucewood, brucewoodd, danceloc, fragranc, mission, lon, itsaboutthework, missouri, mavsvslak	New York Knicks assistant coaches, Free parties, Fashion museums
4	commonwealth, marathon, holland, zaza, patron, valwood, pki, stamped, dnce, typist	Sancti Spíritus, People extradited to Italy, Berlin Marathon

Tabla 5.9: Asignaciones a los *topics* para Dallas (corpus reducido)

Topic	Palabras	Categorías
0	kcvsden, secret, salad, washpark, brock, cliniqu, halsey, snowday,pho, farro	Estée Lauder Companies, Vietnamese soups, Vietnamese noodles
1	forev, sous, starbucksredcup, gust, lakesid, basketbal, pit,applewood, wynkoop, dam	Aviation meteorology, Pacific Gas and Electric Company dams, Prefectures
2	cbe, wcet, floor, dec, draft, regram, lbs, squar, kitten, hidaway	18-bit computers, Online K-12 Schools, Thoroughbred family 1-x
3	pari, load, houseperson, recoveri, listen, journey, fug, renaiss,torylanez, cervantesmasterpiec	Films featuring an item number, Demoscene software, Yachts
4	msw, premium, ldr, vmware, haircut, oyster, document, dress,scorpion, sherri	Italian post-rock groups, Grenadian female sprinters, Cable television

Tabla 5.10: Asignaciones a los *topics* para Denver (corpus reducido)

Topic	Palabras	Categorías
0	divers, thevoic, aceencor, mtv, nbcthevoic, thevoicecast, tragedi,dressbarn, itsi, bitsi	Shakespearean tragedies, Viva Entertainment, Cultural economics
1	favorita, chiqui, pídela, aprovecham, sencillo, fin, pierdan, tendrán,woodworkz, purchas	Television syndication, Arctiina, Arranged marriage
2	entra, faltan, marathon, venetianvega, hibachi, forest, dorado,cantina, messi, yell	The Middle (TV series), Fictional drinking establishments, American college cheerleading squads
3	petra, shia, skywalk, vli, phoenix, pathologist, speech, trichom,adventuredom, vet	Grant Broadcasters, Petra, Plant morphology
4	prayforpari, cheesecak, hobowheewi, wheewinbohovega, hooter,wisdomwednesdaywithchiqui, minion, militari, crap, rang	Celebrity fandom, Despicable Me (franchise), Professional golf tours

Tabla 5.11: Asignaciones a los *topics* para Las Vegas (corpus reducido)

Topic	Palabras	Categorías
0	saban, marchudson, misunderstood, fonda, ahora, rhfajob, vampir,noho, tiki, wreck	Guatemalan cuisine, Films about emotions, Podemos (Spanish political party)
1	devachanla, afifest, fierc, carpet, liver, collag, recreat, devachan,devacurlcno, bradburi	Citation overkill, Uşak, Works about the Holocaust
2	columbus, westlak, alleluia, whiski, doyl, pugsofinstagram, struggl,oakland, approx, dangl	Syntactic transformation, Northwest Los Angeles, Indian whisky
3	dancebar, tcl, sexynight, clean, afifest, parksmakelifebett,plusluncheon, bythesea, exquisitamentetv, gourmand	Tcl programming language family, Software that uses Scintilla, Tsuen Wan
4	mayan, prime, quintil, arclightwomen, djs, itsthewayoumov,itsthewayirol, californiatrip, boat, retir	Trent Reznor, Ancient music, Pension funds

Tabla 5.12: Asignaciones a los *topics* para Los Ángeles (corpus reducido)

Topic	Palabras	Categorías
0	trapper, dior, span, keem, artichok, scotch, hotlinebl, dont, lg,	Middle-earth Half-elven, M*A*S*H, Dior
1	nick, renstarknow, renstarapprov, hakeem, santor, roc, rain, idk,yitzhakrabin, select	Minnesota Greens, Yoruba-speaking people, Mutineers
2	attent, votefornash, camerondalla, urgent,pleeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeas, peoplechoiceaward, jk,takesadamatsutani, hauser, wirth	Clinics, Assassinated Nazis, General practice
3	cinema, balmainxhm, dum, godislovesospreadit, speakeasi, arrest,untz, hut, balmainpari, njpac	Software-defined radio, Military radio systems, Lollipops
4	ecuerda, pued, descarga, leruffo, goodnight, onth, tweet, haz,meatpack, thé	Cacti, Real-time web, Microblogging

Tabla 5.13: Asignaciones a los *topics* para Nueva York (corpus reducido)

Topic	Palabras	Categorías
0	supervis, holla, thoma, height, mckay, hoover, conant, fork, beast,canon	Colgate Rochester Crozer Divinity School faculty, McKay family, ASME Medals
1	azhiphopfestiv, laclipp, jaguar, clipper, kappa, heal, onboard, capit,tue, amanda	Federated Malay States people, Engaged Buddhists, Jaguars
2	terror, lo, tan, startup, officemax, phrase, count, mesatemp, translat,haunt	Business incubators, Syntactic categories, Hunan University alumni
3	retent, ironman, fuel, tonit, council, crust, choos, nomad, multi,imaz	Pies, Napier aircraft engines, RCA Victor singles
4	target, imaz, byrdstheword, racer, raxterrack, wixter, rudyprojectna,notezchri, truminati, jihadi	Jihadism, Windows software, Salafi movement

Tabla 5.14: Asignaciones a los *topics* para Phoenix (corpus reducido)

Topic	Palabras	Categorías
0	sfmusictech, addam, lls, massiv, withmylov, amnesia, bartlett, solid,philanthropi, stormwatch	Widescreen comics, Short comics, Wildstorm Publications titles
1	apparel, legit, ugh, sfmusictech, coder, silicon, dba, gorgonzola,pera, rush	Free QDA software, Perry Ellis International brands, Blue cheeses
2	prayforpari, oakdal, treasuri, johnnicolayphotographi, johnnicolay,chu, weddingphotograph, stress, stanford, juturna	Populated places on the Thames River (Connecticut), Rome R. IX Pigna, Georgia College & State University
3	flame, sanfranciscobayarea, www, moth, noah, lee, mailbox, philip,ggb, avoid	Sorting offices, Kru languages, Nigerian media personalities
4	dpt, naturelov, futurestack, naturaleza, walt, skylov, goldenhour,facebook, setup, wolv	Netherlandish art, Extracellular matrix proteins, Art genres

Tabla 5.15: Asignaciones a los *topics* para San Francisco (corpus reducido)

Topic	Palabras	Categorías
0	prayformizzou, scienceabook, purposetour, commission,ponderosa, ikeepthink, pwis, gopdeb, askamel, hero	Buffet restaurants, Des Moines metropolitan area, Educational robots
1	glamourwoti, morespecificdatingsit, fought, artjamz, analysi, guid,tgsm, wizthund, bei, bombshel	Golf course architects, Guides, Green chemistry
2	streetportrait, emblem, fate, analogphotographi, netanyahu,filmisback, featheredcelebr, peggi, noonan, fleetwood	Israeli medievalists, UNC Greensboro Spartans, Women psychologists
3	oddplacestopropos, draftk, fanduel, tsunami, gave, gopdeb,thorough, fso, nbcwashington, dps	Delhi Public School Society, Superacids, Tonality
4	gopdeb, towjsv, stopitdad, screamqueen, devop, theflash, lhhh,outfieldtonumberon, dwts, bustyink	Agile software development, Dancing with the Stars (U.S. TV series), Software development process

Tabla 5.16: Asignaciones a los *topics* para Washington (corpus reducido)

5.2.2. Corpus completo

Topic	Palabras	Categorías
0	map, legoland, hotchocolaterun, themadnesstour, themakeupshow, religi, southwest, bae, ceremoni, uicrha	Legoland, Miniature parks, BAE Systems facilities
1	actuari, southeast, shedd, greet, gas, internet, lash, fix, liberti, sight	Margarine brands, Actuarial associations, Military discipline
2	reject, nellcôt, corridor, hairstylist, rue, piri, skydeckchicago,lovefashionletstalk, immedi, dbzkai	French Riviera, Fashion occupations, Personal care and service occupations
3	respect, renaiss, trick, velvet, thai, alien, concept, copywrit, bts,puppi	Advertising occupations, Hugo Awards, Communication design
4	campus, best, midwestlegend, yellowcard, scorpio, wermus, portag,themakeupshowchicago, behavior, backstagel	Dray Prescot series, Irony, Christmas short stories

Tabla 5.17: Asignaciones a los *topics* para Chicago

Topic	Palabras	Categorías
0	walnut, crab, growth, chapel, hrh, liveroyali, riverfront, boil, iron, ojo	Maryland cuisine, Anacostia River, Experimental cat breeds
1	pari, prayforpari, franc, terror, rapper, boosi, swim, nomad,fenomenal, soire	Erotic photography, McKinley Senior High School alumni, Napier aircraft engines
2	gogolbordello, momo, princess, np, gypsypunk, clip, doughnut, iren, semifin, trainer	Doughnuts, Fictional ogres, Tournament systems,
3	parad, brucewood, brucewoodd, danceloc, fragranc, mission, lon, itsaboutthework, missouri, mavsvslak	Jennifer Lopez perfumes, Perfumery, Cambodian anti-communists
4	commonwealth, marathon, holland, zaza, patron, valwood, pki, stamped, dnce, typist	Films based on Hungarian novels, White Fathers missions, Office and administrative support occupations

Tabla 5.18: Asignaciones a los *topics* para Dallas

Topic	Palabras	Categorías
0	kcvsden, secret, salad, washpark, brock, cliniqu, halsey, snowday,pho, farro	Inclement weather management, Estée Lauder Companies, Uruguayan stage actresses
1	forev, sous, starbucksredcup, gust, lakesid, basketbal, pit,applewood, wynkoop, dam	Lutherans, Nazarene theologians, English cheeses
2	cbe, wcet, floor, dec, draft, regram, lbs, squar, kitten, hidaway	Broadcast call sign disambiguation pages, PBS member stations, Arab sign languages
3	pari, load, houseperson, recoveri, listen, journey, fug, renaiss,torylanez, cervantesmasterpiec	World War II German radars, Radar warning receivers, HP LaserJet printers
4	msw, premium, ldr, vmware, haircut, oyster, document, dress,scorpion, sherri	Military organization, VMware, Ostreidae

Tabla 5.19: Asignaciones a los *topics* para Denver

Topic	Palabras	Categorías
0	divers, thevoic, aceencor, mtv, nbcthevoic, thevoicecast, tragedi,dressbarn, itsi, bitsi	American nursery rhymes, Companies listed on NASDAQ, Early childhood education
1	favorita, chiqui, pídela, aprovecham, sencillo, fin, pierdan, tendrán,woodworkz, purchas	Modernismo, Jenni Rivera, Cerro Porteño managers
2	entra, faltan, marathon, venetianvega, hibachi, forest, dorado,cantina, messi, yell	Stoves, Companies listed on the Oslo Stock Exchange, Japanese pottery
3	petra, shia, skywalk, vli, phoenix, pathologist, speech, trichom,adventuredom, vet	Mandalay Resort Group, Catskill High Peaks, Indoor amusement parks
4	prayforpari, cheesecak, hobowheewi, wheewinbohovega, hooter,wisdomwednesdaywithchiqui, minion, militari, crap, rang	Meizu, Celebrity fandom, Japanese-American cuisine

Tabla 5.20: Asignaciones a los *topics* para Las Vegas

Topic	Palabras	Categorías
0	saban, marchudson, misunderstood, fonda, ahora, rhfajob, vampir,noho, tiki, wreck	Israeli financial businesspeople, Fonda family, Podemos (Spanish political party)
1	devachanla, afifest, fierc, carpet, liver, collag, recreat, devachan,devacurlcno, bradburi	Theosophy, Medieval Russian people, Horn concertos
2	columbus, westlak, alleluia, whiski, doyl, pugsofinstagram, struggl,oakland, approx, dangl	Masses (music), Approximation theory, Indian whisky
3	dancebar, tcl, sexynight, clean, afifest, parksmakelifebett,pluslunch eon, bythesea, exquisitamentetv, gourmand	Tcl programming language family, Food and drink appreciation, Perfumery
4	mayan, prime, quintil, arclightwomen, djs, itsthewayoumov,itsthewayirol, californiatrip, boat, retir	Metricated units, Contract research organizations, Formula SAE

Tabla 5.21: Asignaciones a los *topics* para Los Ángeles

Topic	Palabras	Categorías
0	trapper, dior, span, keem, artichok, scotch, hotlinebl, dont, lg,	Achill Island, Songs about telephone calls, UK R&B Singles Chart number-one singles
1	nick, renstarknow, renstarapprov, hakeem, santor, roc, rain, idk,yitzhakrabin, select	American male rappers, Minnesota Greens, Images requiring maintenance
2	attent, votefornash, camerondalla, urgent,pleeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeas, peoplechoiceaward, jk,takesadamatsutani, hauser, wirth	Lihula Parish, Clinics, NorthWestern Corporation dams
3	cinema, balmainxhm, dum, godislovesospreadit, speakeasi, arrest,untz, hut, balmainpari, njpac	Akron Pros coaches, Irish television shows, American male hurdlers
4	ecuerda, pued, descarga, leruffo, goodnight, onth, tweet, haz,meatpack, thé	Salsa, Son cubano, LGBT dance

Tabla 5.22: Asignaciones a los *topics* para Nueva York

Topic	Palabras	Categorías
0	supervis, holla, thoma, height, mckay, hoover, conant, fork, beast,canon	Colgate Rochester Crozer Divinity School faculty, Holi, Rupnagar
1	azhiphopfestiv, laclipp, jaguar, clipper, kappa, heal, onboard, capit,tue, amanda	Tea clippers, Jaguars, Software engineering terminology
2	terror, lo, tan, startup, officemax, phrase, count, mesatemp, translat,haunt	Companies formerly listed on the New York Stock Exchange, Business incubators, Illinois
3	retent, ironman, fuel, tonit, council, crust, choos, nomad, multi,imaz	Timex Group, Napier aircraft engines, Hum Sitaray
4	target, imaz, byrdstheword, racer, raxterrack, wixter, rudyprojectna,notezchri, truminati, jihadi	Jihadism, Athletic Bilbao non-playing staff, Anti-Christian sentiment

Tabla 5.23: Asignaciones a los *topics* para Phoenix

Topic	Palabras	Categorías
0	sfmusictech, addam, lls, massiv, withmylov, amnesia, bartlett, solid,philanthropi, stormwatch	Widescreen comics, French Sign Language family, Wildstorm Publications titles
1	apparel, legit, ugh, sfmusictech, coder, silicon, dba, gorgonzola,pera, rush	Blue cheeses, Atlanta Falcons, Comedy tours
2	prayforpari, oakdal, treasuri, johnnicolayphotographi, johnnicolay,chu, weddingphotograph, stress, stanford, juturna	Water goddesses, Rome R. IX Pigna, Cossidae
3	flame, sanfranciscobayarea, www, moth, noah, lee, mailbox, philip,ggb, avoid	United Parcel Service, Sorting offices, Kru languages
4	dpt, naturelov, futurestack, naturaleza, walt, skylov, goldenhour,facebook, setup, wolv	Netherlandish art, Sony, Cisco protocols

Tabla 5.24: Asignaciones a los *topics* para San Francisco

Topic	Palabras	Categorías
0	prayformizzou, scienceabook, purposetour, commission,ponderosa, ikeepthink, pwis, gopdeb, askamel, hero	Mandarina, Lemons, Woodboring beetles
1	glamourwoti, morespecificdatingsit, fought, artjamz, analysi, guid,tgsm, wizthund, bei, bombshel	Hugo Award for Best Non-Fiction Book winning works, Works about Marilyn Monroe, Feminist essays
2	streetportrait, emblem, fate, analogphotographi, netanyahu,filmisback, featheredcelebr, peggi, noonan, fleetwood	Israeli medievalists, UNC Greensboro Spartans, Israeli business executives
3	oddplacestopropos, draftk, fanduel, tsunami, gave, gopdeb,thorough, fso, nbcwashington, dps	Fantasy sports, Browser-based game websites, Delhi Public School Society
4	gopdeb, towjsv, stopitdad, screamqueen, devop, theflash, lhhh,outfieldtonumberon, dwts, bustyink	Agile software development, Dancing with the Stars (U.S. TV series), Software development process

Tabla 5.25: Asignaciones a los *topics* para Washington

Debido a las discrepancias en las métricas entre este proyecto y el de Schönhofen [30], es posible que las categorías asignadas a los *topics* no sean del todo correctas, es por ello que en algunos casos parece que no exista relación y en otros sí.

CAPÍTULO 6
Conclusiones

En este proyecto se ha implementado un sistema que identifique los temas y asigne una categoría a los temas que se hablan en los tuits de una ciudad. Para ello se ha realizado la limpieza y procesado de tuits aplicando el algoritmo de LDA. También para cumplir dicho objetivo, se ha creado un sistema de clasificación en categorías usando la información contenida en la Wikipedia en inglés, para después aplicarlo sobre la salida de LDA. Se ha realizado un análisis del clasificador empleando los propios artículos de Wikipedia y posteriormente se han clasificado los *topics* obtenidos tras emplear LDA sobre los tuits de un conjunto de ciudades de EE.UU.

Tal y como se ha visto en el apartado de los resultados (sección 5.1.3), los resultados obtenidos con el clasificador no se equiparan a los de Schönhofen[30] pero sin embargo esto se puede deber a diversos motivos.

Desde un principio de ha establecido que el objetivo era crear una herramienta que sirviese de ayuda (ver sección 1.2) más que realizar un clasificador perfecto de documentos. Del mismo modo que un traductor automático como pueda ser el de Google[1] no realiza un trabajo perfecto (aunque en los últimos años ha mejorado considerablemente), si que se obtiene un resultado desde el cuál partir, acelerando el proceso.

Se ha podido ver cómo se clasificaban los temas obtenidos por LDA (ver sección 5.2) y aunque no todas las categorías sugeridas están relacionadas entre sí siempre, el usuario puede hacerse una idea, facilitando con qué categoría etiquetar el *topic*.

Debido a las limitaciones de tiempo y de las máquinas donde se ha ejecutado el clasificador, no se han podido realizar todas las pruebas que se habían pensado en un principio, ya que no sólo no se ha podido hacer una validación cruzada completa, tampoco se han podido probar otras configuraciones en el filtrado de categorías.

6.1 Trabajos futuros

Los trabajos futuros del presente proyecto estarían enfocadas a evitar las limitaciones encontradas, se podría buscar una forma de paralelizar el entrenamiento o la evaluación siempre y cuando se encontrase una forma de trabajar con un gran volumen de datos en memoria secundaria de forma más eficiente que con *Shove*. Una posible solución podría ser *Dask*[2], que se oferta como una forma fácil y sencilla de paralelizar procesos y grandes volúmenes de datos incluso en un sólo equipo.

[1] `https://translate.google.com/`
[2] `https://dask.pydata.org`

De forma consecuente, si se mejorasen los tiempos de ejecución, sería posible realizar pruebas variando qué categorías se quedan fuera ya sea por palabras clave o por número de artículos asociados.

Otro trabajo futuro sería implementar un árbol ontológico de categorías como en [25] con el fin de generalizar la categorías etiquetadas en el sistema entrenado y así clasificar según unas categorías más generales y descriptivas del tema tratado.

Bibliografía

[1] *We're all atwitter: 3 times President Trump made major announcements via tweets.* 2018. URL: https://usat.ly/2pd4x4H (visitado 14-05-2018).

[2] *Twitter is becoming the first and quickest source of investment news.* URL: https://www.theguardian.com/technology/2013/apr/23/twitter-first-source-investment-news (visitado 03-07-2018).

[3] Aliza Sarlan, Chayanit Nadam y Shuib Basri. «Twitter sentiment analysis». En: *Information Technology and Multimedia (ICIMU), 2014 International Conference on*. IEEE. 2014, págs. 212-216.

[4] *Global software supplier SAP positions a new product with Twitter targeting.* URL: https://marketing.twitter.com/na/en/success-stories/global-software-supplier-sap-positions-a-new-product.html (visitado 14-05-2018).

[5] *Twitter Campaign analytics.* URL: https://marketing.twitter.com/na/en/solutions/measure-results/campaign-analytics.html (visitado 14-05-2018).

[6] *Audience targeting.* URL: https://marketing.twitter.com/na/en/solutions/create-engagement/audience-targeting.html (visitado 14-05-2018).

[7] Daniel M. Romero, Brendan Meeder y Jon Kleinberg. «Differences in the Mechanics of Information Diffusion Across Topics: Idioms, Political Hashtags, and Complex Contagion on Twitter». En: *Proceedings of the 20th International Conference on World Wide Web*. WWW '11. Hyderabad, India: ACM, 2011, págs. 695-704. ISBN: 978-1-4503-0632-4. DOI: 10.1145/1963405.1963503. URL: http://doi.acm.org/10.1145/1963405.1963503.

[8] Elena del Val, Javier Palanca y Miguel Rebollo. «U-Tool: A Urban-Toolkit for Enhancing City Maps Through Citizens' Activity». En: *Advances in Practical Applications of Scalable Multi-agent Systems. The PAAMS Collection*. Ed. por Yves Demazeau y col. Cham: Springer International Publishing, 2016, págs. 243-246. ISBN: 978-3-319-39324-7.

[9] Elizabeth Vivanco y col. «Using Geo-Tagged Sentiment to Better Understand Social Interactions». En: *Advances in Practical Applications of Cyber-Physical Multi-Agent Systems: The PAAMS Collection*. Ed. por Yves Demazeau y col. Cham: Springer International Publishing, 2017, págs. 369-372. ISBN: 978-3-319-59930-4.

[10] E. del Val, C. Martínez y V. Botti. «A Multi-agent Framework for the Analysis of Users Behavior over Time in On-Line Social Networks». En: *10th International Conference on Soft Computing Models in Industrial and Environmental Applications*. Ed. por Álvaro Herrero y col. Cham: Springer International Publishing, 2015, págs. 191-201. ISBN: 978-3-319-19719-7.

[11] Liliber Alvarez Ramos. «Análisis de ciudades a través su actividad en redes sociales». En: (2017).

[12] Bodong Chen, Xin Chen y Wanli Xing. «"Twitter Archeology" of Learning Analytics and Knowledge Conferences». En: *Proceedings of the 5th International Conference on Learning Analytics and Knowledge, LAK 2015*. Association for Computing Machinery. 2015, págs. 340-349.

[13] David Alfred Ostrowski. «Using latent dirichlet allocation for topic modelling in twitter». En: *Semantic Computing (ICSC), 2015 IEEE International Conference on*. IEEE. 2015, págs. 493-497.

[14] David M Blei, Andrew Y Ng y Michael I Jordan. «Latent dirichlet allocation». En: *Journal of machine Learning research* 3.Jan (2003), págs. 993-1022.

[15] *Iris flower data set*. URL: `https://en.wikipedia.org/wiki/Iris_flower_data_set` (visitado 14-05-2018).

[16] Roberto Lopez. *Iris flowers classification*. URL: `https://www.neuraldesigner.com/learning/examples/iris_flowers_classification` (visitado 14-05-2018).

[17] Marıa Luz Congosto, Montse Fernández y Esteban Moro. «Twitter y polıtica: Información, opinión y?‘ Predicción?» En: (2011).

[18] Hao Wang y col. «A system for real-time twitter sentiment analysis of 2012 us presidential election cycle». En: *Proceedings of the ACL 2012 System Demonstrations*. Association for Computational Linguistics. 2012, págs. 115-120.

[19] Shashank Gupta. *Sentiment Analysis: Concept, Analysis and Applications*. URL: `https://towardsdatascience.com/sentiment-analysis-concept-analysis-and-applications-6c94d6f58c17` (visitado 14-05-2018).

[20] Bo Hu y Martin Ester. «Spatial topic modeling in online social media for location recommendation». En: *Proceedings of the 7th ACM conference on Recommender systems*. ACM. 2013, págs. 25-32.

[21] Guy Lansley y Paul A Longley. «The geography of Twitter topics in London». En: *Computers, Environment and Urban Systems* 58 (2016), págs. 85-96.

[22] *Harmonic mean*. URL: `https://en.wikipedia.org/wiki/Harmonic_mean` (visitado 14-05-2018).

[23] João Pereira y col. «Characterizing geo-located tweets in brazilian megacities». En: (2017).

[24] N Grinberg y col. «Extracting diurnal patterns of real world activity from social media». En: (ene. de 2013), págs. 205-214.

[25] Mostafa M. Hassan, Fakhri. Karray y Mohamed. S. Kamel. «Automatic Document Topic Identification using Wikipedia Hierarchical Ontology». En: *2012 11th International Conference on Information Science, Signal Processing and their Applications (ISSPA)*. 2012, págs. 237-242. DOI: `10.1109/ISSPA.2012.6310552`.

[26] Chandler May y col. «Topic Identification and Discovery on Text and Speech». En: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (ene. de 2015), págs. 2377-2387.

[27] Hanna M Wallach. «Topic modeling: beyond bag-of-words». En: *Proceedings of the 23rd international conference on Machine learning*. ACM. 2006, págs. 977-984.

[28] Benno Stein y Sven Meyer. «Topic Identification: Framework and Application». En: 2004.

[29] Yingjie Lu. «Automatic topic identification of health-related messages in online health community using text classification». En: *SpringerPlus*. 2013.

[30] Peter Schonhofen. «Identifying Document Topics Using the Wikipedia Category Network». En: *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence*. IEEE Computer Society, 2006, págs. 456-462. ISBN: 0-7695-2747-7.

[31] Chin-Yew Lin. «Knowledge-based Automatic Topic Identification». En: *Proceedings of the 33rd Annual Meeting on Association for Computational Linguistics*. ACL '95. Cambridge, Massachusetts: Association for Computational Linguistics, 1995, págs. 308-310. DOI: 10.3115/981658.981705. URL: https://doi.org/10.3115/981658.981705.

[32] R. Karim. *Scala Machine Learning Projects*. Packt Publishing, 2018. ISBN: 9781788479042. URL: https://books.google.es/books?id=xnaAswEACAAJ.

[33] Rianne Kaptein y Jaap Kamps. «Exploiting the category structure of Wikipedia for entity ranking». En: *Artificial Intelligence* 194 (2013). Artificial Intelligence, Wikipedia and Semi-Structured Resources, págs. 111-129. ISSN: 0004-3702. DOI: https://doi.org/10.1016/j.artint.2012.06.003. URL: http://www.sciencedirect.com/science/article/pii/S0004370212000732.

[34] Jin-Xia Huang y col. «Extract Reliable Relations from Wikipedia Texts for Practical Ontology Construction». en. En: *ComputaciÃy Sistemas* 20 (sep. de 2016), págs. 467-476. ISSN: 1405-5546. URL: http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1405-55462016000300467&nrm=iso.

[35] Torsten Zesch, Iryna Gurevych y Max MÃhlhÃ¤user. «Analyzing and Accessing Wikipedia as a Lexical Semantic Resource». En: (ene. de 2007).

[36] Evgeniy Gabrilovich y Shaul Markovitch. «Computing Semantic Relatedness Using Wikipedia-based Explicit Semantic Analysis». En: *Proceedings of the 20th International Joint Conference on Artifical Intelligence*. IJCAI'07. Hyderabad, India: Morgan Kaufmann Publishers Inc., 2007, págs. 1606-1611. URL: http://dl.acm.org/citation.cfm?id=1625275.1625535.

[37] *Schwartzian transform*. URL: https://en.wikipedia.org/wiki/Schwartzian_transform (visitado 05-07-2018).

[38] *WordNet Domains*. URL: http://wndomains.fbk.eu/index.html (visitado 20-12-2017).

[39] *Inodo*. URL: https://es.wikipedia.org/wiki/Inodo (visitado 15-05-2018).

APÉNDICE A
Resultados completos

En este apéndice se mostrarán todos los resultados obtenidos para todos los temas, tanto seleccionando las palabras para LDA como en el clasificador. En cada listado el orden es de mayor a menor peso. Los resultados con todos sus pesos asociados se encuentran en el repositorio de GitHub[1].

[1] `https://github.com/carlos3dx/tfm2018`

A.1 LDA

A continuación se muestran las diez palabras de mayor peso para cada *topic* de cada ciudad según el algoritmo LDA.

Topic	Palabras
0	map, legoland, hotchocolaterun, themadnesstour, themakeupshow, religi, southwest, bae, ceremoni, uicrha
1	actuari, southeast, shedd, greet, gas, internet, lash, fix, liberti, sight
2	reject, nellcôt, corridor, hairstylist, rue, piri, skydeckchicago, lovefashionletstalk, immedi, dbzkai
3	respect, renaiss, trick, velvet, thai, alien, concept, copywrit, bts, puppi
4	campus, best, midwestlegend, yellowcard, scorpio, wermus, portag, themakeupshowchicago, behavior, backstagel
5	willrunforchocol, regular, hotchocolaterac, sweetest, rampag, resid, graceland, outback, soho, strateg
6	oiler, simon, judg, edmonton, chili, harri, sticker, toll, romanian, porter
7	liegt, wo, quiz, madnesstour, scorpion, malnati, eleven, ramenfest, alon, restaurantjob
8	box, tinley, soulcycl, chug, ignit, rite, bishopford, sibley, blvd, standard
9	ot, japanes, consid, tilt, slept, ceo, tele, tantrum, rhythm, hostess
10	penn, timeout, suga, dian, phylli, wife, acadia, shoutout, afford, blackbird
11	portag, hansen, gunnar, william, pipelin, keyston, nix, laser, hazemaz, colombian
12	christmasdecor, xmas, christmaslight, reunit, reviv, werk, bistro, geek, boybandreview, particip
13	highway, labandal, trailer, strike, jaim, cruz, glori, pictureperfect, nhl, cranberri
14	bishop, prn, dawg, barr, decor, tn, walkforlittlec, cityandcolour, burr, elev

Tabla A.1: Salida LDA para Chicago

Topic	Palabras
0	walnut, crab, growth, chapel, hrh, liveroyali, riverfront, boil, iron, ojo
1	pari, prayforpari, franc, terror, rapper, boosi, swim, nomad, fenomenal, soire
2	gogolbordello, momo, princess, np, gypsypunk, clip, doughnut, iren, semifin, trainer
3	parad, brucewood, brucewoodd, danceloc, fragranc, mission, lon, itsaboutthework, missouri, mavsvslak
4	commonwealth, marathon, holland, zaza, patron, valwood, pki, stamped, dnce, typist
5	brg, laker, regul, expert, bid, artcon, sportsday, burlesqu, anybodi, teammat
6	captain, doughnut, cbd, generous, newman, parcel, ssis, anatol, alarm, haunt
7	brewer, smdalla, rye, varagesal, banquet, niagara, payrol, ink, jalapeño, investmentbank
8	shut, mariachielbronx, gratitud, spiritu, elmstfest, fork, cousin, bullshit, brewersbal, bronx
9	brewer, western, lmaoooo, et, tectum, elix, manim, dime, dun, sambuca
10	simpl, gross, molli, istandarddalla, steel, ped, fm, schnitzer, ind, inventori
11	heard, envi, height, divi, peterpan, speed, suspici, militari, dallascowboy, midnight
12	presentado, tarima, rompiendo, ra, happyveteransday, pensk, flow, soup, chair, feb
13	jet, winnipeg, grey, cath, gritz, casual, rc, bruh, routin, lincoln
14	dallasstar, invit, owner, demdeb, consid, cattl, toronto, awar, conserv, healthwelfar
15	gift, ursulin, air, miller, safe, dayl, sonus, craftsman, mix, flexibl

Tabla A.2: Salida LDA para Dallas

Topic	Palabras
0	kcvsden, secret, salad, washpark, brock, cliniqu, halsey, snowday, pho, farro
1	forev, sous, starbucksredcup, gust, lakesid, basketbal, pit, applewood, wynkoop, dam
2	cbe, wcet, floor, dec, draft, regram, lbs, squar, kitten, hidaway
3	pari, load, houseperson, recoveri, listen, journey, fug, renaiss, torylanez, cervantesmasterpiec
4	msw, premium, ldr, vmware, haircut, oyster, document, dress, scorpion, sherri
5	trey, espresso, neuro, mission, wife, spectra, mgmt, reminisc, lonetre, beverag
6	franc, urgent, cannabi, civil, principaladdress, hoodi, violenc, given, hipster, twfsl
7	bou, spotter, champion, thisrunnerlif, mustachedach, lifeisshortrunhappi, former, rye, yrs, csa
8	river, kcvsden, other, marianastrench, butterfli, nicu, hoo, woo, limon, adoni
9	counter, dfw, castl, csp, qmap, dress, biker, alright, tv, resid
10	turnip, prayforpari, melt, gayfollow, gopenskecar, boutiqu, ibak, element, coq, sportif
11	dfw, tracksuitwed, marin, io, kim, lowri, cheerlead, pensk, raid, fedex
12	corp, healthon, wild, pari, rabbit, hologram, foamposit, westminist, holli, rousey
13	money, josh, monkey, cri, ele, jam, exercis, minibandmonday, requir, gel

Tabla A.3: Salida LDA para Denver

Topic	Palabras
0	divers, thevoic, aceencor, mtv, nbcthevoic, thevoicecast, tragedi, dressbarn, itsi, bitsi
1	favorita, chiqui, pídela, aprovecham, sencillo, fin, pierdan, tendrán, woodworkz, purchas
2	entra, faltan, marathon, venetianvega, hibachi, forest, dorado, cantina, messi, yell
3	petra, shia, skywalk, vli, phoenix, pathologist, speech, trichom, adventuredom, vet
4	prayforpari, cheesecak, hobowheewi, wheewinbohovega, hooter, wisdomwednesdaywithchiqui, minion, militari, crap, rang
5	respiratori, root, bigtruck, mozen, proof, bean, summit, endomondo, rockin, sake
6	stripatnight, safeti, exaclibur, tocaucus, skyloft, gmg, runhappi, lvrnr, cortez, basebal
7	shoulder, bear, ryanlong, playboy, inevan, sir, alt, doübl, sleigh, calvinharri
8	stripatnight, dean, salonclosetothestrip, celebritystylistprincessleah, mcdonald, nitto, stylistformenandwomen, muerto, ferri, crab
9	bonus, prayer, jackpot, omnianightclub, bigsmok, eiffeltow, womengrow, casinoio, freespin, appstoregam
10	cri, clubhishop, fightfor, ascend, hishopn, chili, grow, raisethewag, immigr, theun
11	victim, thevenetian, terror, vegasgroup, velveteen, rabbit, dim, thenewaveng, bossbe, lip
12	escap, vegaswithaaron, northeast, teamdangl, dpi, outstand, hoe, cri, clip, groom
13	pierdan, carhop, djlaros, canal, malley, jenniv, sabado, artsdistrictlv, usmc, deck
14	mjbizcon, cannabi, boulder, angelinvest, stripatnight, law, mojito, headset, ambassador, captur

Tabla A.4: Salida LDA para Las Vegas

Topic	Palabras
0	saban, marchudson, misunderstood, fonda, ahora, rhfajob, vampir, noho, tiki, wreck
1	devachanla, afifest, fierc, carpet, liver, collag, recreat, devachan, devacurlcno, bradburi
2	columbus, westlak, alleluia, whiski, doyl, pugsofinstagram, struggl, oakland, approx, dangl
3	dancebar, tcl, sexynight, clean, afifest, parksmakelifebett, pluslunchcon, bythesea, exquisitamentetv, gourmand
4	mayan, prime, quintil, arclightwomen, djs, itsthewayoumov, itsthewayirol, californiatrip, boat, retir
5	shea, nicolett, nationaldonutday, rewrit, nicoletteshea, kidz, pandora, ting, mentorship, stem
6	marriott, mortal, envivo, gamestop, blackop, , fbf, saúlhernández, rhmrjob, ralph
7	tgif, ohm, liquor, venu, feat, stunt, util, holder, kingdiamond, seek
8	millionmaskmarch, mmmla, doubletal, sullen, sullenmodel, sullenangel, analyt, irvin, lafunghi, beethoven
9	small, ballroom, teragram, southeastern, jiujitsu, galleria, transit, nationalredheadday, squat, pueden
10	minion, energi, saulhernandez, spread, server, opm, lesson, perch, pca, mamatit
11	sheat, goldstein, snapback, autograph, woodworkz, practition, sbcr, gator, redsox, sheraton
12	redcat, edna, calart, roy, ultra, zaman, mark, eo, ron, clean

Tabla A.5: Salida LDA para Los Ángeles

Topic	Palabras
0	trapper, dior, span, keem, artichok, scotch, hotlinebl, dont, lg,
1	nick, renstarknow, renstarapprov, hakeem, santor, roc, rain, idk, yitzhakrabin, select
2	attent, votefornash, camerondalla, urgent, pleeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeas, peoplechoiceaward, jk, takesadamatsutani, hauser, wirth
3	cinema, balmainxhm, dum, godislovesospreadit, speakeasi, arrest, untz, hut, balmainpari, njpac
4	recuerda, pued, descarga, leruffo, goodnight, onth, tweet, haz, meatpack, thé
5	nashgrier, thé, gillian, artistri, thecutlif, garcia, cloudi, systhem, captain, westsid
6	ladybabi, brazil, sob, ladybeard, mapl, tunl, churchflow, refrescatumundo, thé, effen
7	ward, mic, martini, bff, zumba, toi, cma, attack, showcas, stranger
8	hereditari, reced, hairlin, unispher, waffl, adida, incausa, letitplayahhgotchu, howl, sketchi
9	parang, cleopatra, daylinlainemua, tranni, thedoubleup, madeintheeast, mite, backstag, invent, monro
10	openhous, projectwildrabbit, noa, shaggi, amc, imax, sonicmotionpictur, thé, rodger, carterreynold
11	isi, eatali, prom, recap, happiest, allerton, entranc, sangria, roxi, beatmak
12	joe, cmaaward, carnegiehal, along, precious, davidlynch, encor, ballerbrownsworld, mercuri, karen
13	bang, boom, curious, konditionsportsmedia, truefan, dumont, arbi, onthemajordeeganexpwi, miller, supernatur
14	nightclub, appoint, pieprz, guadalup, honn, sub, chop, irish, soda, ellen
15	crowd, onthecrossbronxexpressway, primit, strapback, damnson, felin, statueofliberti, firewat, ualreadysnowwww, raison
16	ramen, ontheb, mcguiness, totto, fame, ash, walker, skillman, esm, housem
17	changebeginswithin, lift, becam, feinstein, gen, weigh, stylist, printfair, chillin, grain
18	knick, wild, ballet, veggi, vídeo, barstar, iamjojotour, palac, pile, onion

Tabla A.6: Salida LDA para Nueva York

Topic	Palabras
0	supervis, holla, thoma, height, mckay, hoover, conant, fork, beast, canon
1	azhiphopfestiv, laclipp, jaguar, clipper, kappa, heal, onboard, capit, tue, amanda
2	terror, lo, tan, startup, officemax, phrase, count, mesatemp, translat, haunt
3	retent, ironman, fuel, tonit, council, crust, choos, nomad, multi, imaz
4	target, imaz, byrdstheword, racer, raxterrack, wixter, rudyprojectna, notezchri, truminati, jihadi
5	jumpman, lemon, item, attent, recreat, herb, spacious, sweeti, heartbreak, rp
6	salut, bridg, victori, seahawk, sundevil, vinyl, uwvsasu, roomi, medicin, cub
7	shupsquad, ironman, genuin, flip, peoria, arizonatattoo, bevmo, northeast, preserv, puresaturday
8	scottsdalebar, scottsdalenightlif, imaz, intlsundaynight, thunderbird, intlsaturdaynight, airway, azvssea, disgrac, westbound
9	edmfamili, edcfamili, bass, edcazfam, edmlif, lt, localfirst, marcus, holm, outdoor
10	preserv, punch, core, orangetheori, maker, formal, dosimetrist, coverag, nightmar, azhiphopfestiv
11	parisattack, annex, explos, assault, medicaldevic, climb, ex, compens, barrel, kbb
12	sunsvsclipp, skillsoft, demdeb, finger, beatla, ribbon, cane, nagc, feelthebern, ot
13	crab, mckellip, iya, siriusxm, shack, suppli, brian, tractor, port, destin
14	ironman, laker, gopdeb, imaz, browni, aec, batter, shadow, furious, stall
15	cowgirlzenphotographi, ventsmagazin, eastbound, joel, entertainmentjournalist, ghost, froyo, evan, manor, thecreativegroup
16	cinema, divis, respect, cajunrabbitvintag, cajunrabbit, slang, pantri, inselli, lobbi, turfparadis

Tabla A.7: Salida LDA para Phoenix

Topic	Palabras
0	sfmusictech, addam, lls, massiv, withmylov, amnesia, bartlett, solid, philanthropi, stormwatch
1	apparel, legit, ugh, sfmusictech, coder, silicon, dba, gorgonzola, pera, rush
2	prayforpari, oakdal, treasuri, johnnicolayphotographi, johnnicolay, chu, weddingphotograph, stress, stanford, juturna
3	flame, sanfranciscobayarea, www, moth, noah, lee, mailbox, philip, ggb, avoid
4	dpt, naturelov, futurestack, naturaleza, walt, skylov, goldenhour, facebook, setup, wolv
5	map, hole, nick, provis, iza, cooper, xmed, gracia, trick, bun
6	storm, hostag, graffitti, icymi, futurestack, earring, cellarmak, cancel, breakthroughpr, cortland
7	veteransday, wherein, circasurv, instacool, igcalifornia, willi, especi, patrick, toro, moi
8	ricki, comedian, standup, standupcomedi, lightn, topher, forhomesweetiehom, thunder, l, dmx
9	jacquelin, joie, core, vivr, odd, keyboard, chain, muscl, thrash, thewalkingdead
10	factori, cheesecak, must, background, notif, gough, sunsetlov, skylov, golf, rxbandit
11	codeword, sd, seaworld, orca, icu, latin, veteransday, wonoloapp, pho, diner
12	thoma, pollicita, goodstart, showtimepbbsalubong, musictech, nowhir, sfmusictech, westfield, cabin, vr

Tabla A.8: Salida LDA para San Francisco

Topic	Palabras
0	prayformizzou, scienceabook, purposetour, commission, ponderosa, ikeepthink, pwis, gopdeb, askamel, hero
1	glamourwoti, morespecificdatingsit, fought, artjamz, analysi, guid, tgsm, wizthund, bei, bombshel
2	streetportrait, emblem, fate, analogphotographi, netanyahu, filmisback, featheredcelebr, peggi, noonan, fleetwood
3	oddplacestopropos, draftk, fanduel, tsunami, gave, gopdeb, thorough, fso, nbcwashington, dps
4	gopdeb, towjsv, stopitdad, screamqueen, devop, theflash, lhhh, outfieldtonumberon, dwts, bustyink
5	daystillpurpos, happyveteransday, housew, mondaysin, mondaymotiv, freer, kd, crash, racejustic, vet
6	philosoph, ravenel, odesza, hamilton, thevoic, cbre, swwashingtondc, whether, tsg, delano
7	canwebringback, nerdbroadway, purposetour, dayuntilmitam, tootsi, protect, justinonnova, wreck, dj, rosslyn
8	bebravein, mondaymotiv, badnamesforarockband, askdemi, classic, raider, stream, ncaa, willi, howardtheatr
9	newseum, biafra, fisher, futuredecod, newseumnight, shine, debat, grandopen, hookahexperiencedc, maker
10	nextgenh, untyingtheknot, lls, dori, wwhl, hunter, wet, jfnaga, cross, cosmetolog
11	jane, abandon, object, poppin, ban, vivica, undertak, funk, mission, spam
12	tommi, hanson, mypostapocalypticplan, dongrimmi, underwearsong, ourgenerationourchoic, semperfi, lucado, fightfor, ruinaholidayin
13	matt, cassel, fdic, happierholidaysin, coder, dez, hick, quantico, banner, donni
14	amc, gap, yep, basement, aung, zero, aint, gratitud, joel, creep
15	geffen, webondedov, foxbusinessdeb, endoftheday, dirk, undertak, sinjar, loveyourself, wya, séc
16	rey, blackoncampus, prep, steve, bnaskdanandphil, thunder, youarenotmeanttobeif, gwar, daystilikwydl, selfieforseb
17	lamarr, hedi, cowboy, thewalkingdead, rhoa, rctid, nerdbroadway, dobb, lou, jonathan

Tabla A.9: Salida LDA para Washington

A.2 Clasificación

A continuación se muestran las diez categorías de mayor peso para cada uno de los *topics* de LDA para cada ciudad.

A.2.1. Corpus reducido

Topic	Categorías
0	Legoland, Eritrean culture, Royal Saudi Air Force, Floristic provinces, Hawker Siddeley aircraft, Religion and violence, Ceremonial magic, Map types, Western Finland Province, Christianity
1	Dutch female rowers, Insurance, Fax software, Whipping, Applied statistics, Short stories, European deities, English silent film actresses, Mesoamerican art, American road movies
2	French Riviera, Thoroughbred family 2-e, Sapindales families, Executed Turkish people, Darjeeling district, Fictional association football television series, Share trading, Geopolitical corridors, British teen sitcoms, Hot sauces
3	Advertising occupations, Non-fiction books about acting, Journalism occupations, Belgian architecture, Puppis, Consulting, Folk rock, Communication design, Turntable video games, Gerromorpha
4	Nick Fury, Kursk submarine disaster, Robotic submarines, Greater Prince George, Campuses, Remotely operated underwater vehicles, Dinosaur paleobiology, Portages, Publications, Mind–body problem
5	East Baltimore, Outback Bowl, Latter Day Saint universities and colleges, Minimalism, Florida Gators football bowl games, Southeast Baltimore, Wisconsin Badgers football bowl games, Solomon R. Guggenheim Foundation, Road rallying, Hellcat Records artists
6	Oilers Entertainment Group, Stickers, Crunk, Islands District, American football equipment, Pacific Division (NHL), Romanian Television, Shanghai cuisine, Japan Post Holdings, Thripidae
7	Silurian arthropods, Japanese game shows, Protein toxins, Nothing Records artists, Films about games, Israeli environmentalists, Tank destroyers, Beethoven scholarship, Educational games, Secondary education
8	Sephardi Jewish cuisine, American male triathletes, Xbox 360 software, Yemeni cuisine, Australian Army officers, English ballerinas, Hot sauces, Aquileia, Ignition systems, Massachusetts Democratic-Republicans

Tabla A.10: Asignaciones a los *topics* para Chicago (corpus reducido), parte 1

Topic	Categorías
9	Hostess Brands brands, Elton John, American snack foods, Analysis, Djurgårdens IF Fotboll, British television documentaries, Thioamides, American alcoholic drinks, Hammarby Fotboll, Peripatetic philosophers
10	CONFIG.SYS directives, Canadian female songwriters, English madrigals, Bauxite mines, Cherokee artists, Canadian radio hosts, Telecommunications engineering, American female winemakers, Aimee Mann, Kamen Rider
11	Light bombers, Energy infrastructure under construction, Mountains on Mars, Swedish geneticists, Japanese Latin American, Greater Prince George, TransCanada Corporation, German Christian socialists, Laser types, Triviidae
12	North Korea, Canadian film remakes, North Korea–South Korea border, Abbreviations, Existential risk, National unifications, Třinec, North Korea–South Korea relations, Films about child abuse, Artificial intelligence publications
13	Trailers, Appalachian bogs, Cambridge College alumni, NHL (video game series), Films set during the Philippine American War, Victoria Beckham, Dominican Republic painters, NHL outdoor games, National Hockey League labor relations, Tank transporters
14	Anesthesiology and palliative medicine journals, Trusses, French European Commissioners, String data structures, Service occupations, Perry Mason, Spider anatomy, Shabbat innovations, Graph data structures, Ulster nationalists

Tabla A.11: Asignaciones a los *topics* para Chicago (corpus reducido), parte 2

Topic	Categorías
0	Surat, Hot springs, Kolkata, Anomura, Invasive animal species, Chapels, Postindustrial society, Calappoidea, Peak oil books, Bulgarian noble titles
1	Napier aircraft engines, McKinley Senior High School alumni, Flat engines, Paris Métro rolling stock, Converts to Sunni Islam, Uzbekistani male cyclists, Auto races, Thoroughbred family 1-e, Swimming styles, History books about France
2	Brazilian Carnival, United States intelligence operations, Tournament systems, Swale, Complexity classes, Northwest Region (Cameroon), Pennsylvania Dutch cuisine, Large-group awareness training, Tibetan cuisine, Actinides
3	New York Knicks assistant coaches, Free parties, Fashion museums, American record charts, Painted statue public art, Olfaction, Atlanta Hawks head coaches, Mining culture and traditions, National Cycle Routes, Toiletry
4	Sancti Spíritus, People extradited to Italy, Berlin Marathon, Incidents during the Hajj, Westminster system parliaments, Mexican distilled drinks, Camorristi, King Edward VII-class battleships, World Marathon Majors, Private equity firms
5	Bible, Bible translations into English, Environmentalists, Satire, Arizona Diamondbacks coaches, Heresy, Los Angeles Lakers, Skills, Erotic dance, Regulation
6	United States intelligence operations, Joseph Beuys, CBC Radio One stations, Moscow International Business Center, Pennsylvania Dutch cuisine, Buffalo Sabres draft picks, Electoral Bloc Democratic Moldova MPs, Phi Lambda Upsilon, Alarms, Cariban languages
7	Withholding taxes, Starbucks people, Dutch inventions, César Awards, Rye-based drinks, Banking terms, Walmart people, Payroll, V2 Records EPs, Inks
8	Raw foodism, Uptown Records singles, Injection molding, Environmentalism and religion, Fictional ensigns, Bingo, Federal Assembly (Switzerland), Exxon Valdez oil spill, Starfleet ensigns, New York City opera companies
9	Italian liqueurs, Elaphidion, Psychedelic rock record labels, Shooters (drinks), Staphylinidae, Anise liqueurs and spirits, Diners, Starbucks people, Coll, French musical groups
10	Medical specialties, British female triathletes, Madang languages, Bad Doberan, Inventory, Minicomputers, Income taxation, Interurban railways, ASTM standards, Working capital management

Tabla A.12: Asignaciones a los *topics* para Dallas (corpus reducido), parte 1

Topic	Categorías
11	Channel One (UK TV channel) television programmes, Brazilian composers, English pop guitarists, Hewlett-Packard products, Caesalpinieae, Nerdist Industries, Screamo musical groups, English jazz guitarists, Male composers, Mystery games
12	American Motors, Workplace violence, Honda Formula One cars, February, Vegetable soups, Chairs, Can-Am entrants, Beehives, Italian soups, American racecar constructors
13	Christian Identity, Auckland Regional Councillors, Vietnam War POW/MIA issues, Software optimization, IAI aircraft, Dress codes, Co-operative Commonwealth Federation, Auckland City Councillors, Rover vehicles, Converts to Mormonism
14	Transhumance, Religious philosophy, Thoroughbred family 6-e, Arnold Palmer, Beef cattle breeds, Anarchist organizations, Toronto Indoor, Cattle breeds, Financial capital, Grand Bell Awards
15	Women rabbis, LGBT short story collections, Audio amplifier manufacturers, Remittances, Arts and Crafts architecture, Association football venues, Nonconvex polyhedra, Pop-culture neologisms, Loudspeaker manufacturers, Supernatural

Tabla A.13: Asignaciones a los *topics* para Dallas (corpus reducido), parte 2

Topic	Categorías
0	Estée Lauder Companies, Vietnamese soups, Vietnamese noodles, Cosmetics brands, Tuna dishes, Salads, Soltau, Pingry School alumni, Secret Story, Vietnamese words and phrases
1	Aviation meteorology, Pacific Gas and Electric Company dams, Prefectures, Citizen Kane, Wichita Falls metropolitan area, East German female rowers, Grunge, Dutch cuisine, Atmospheric dynamics, Full Moon Records singles
2	18-bit computers, Online K-12 Schools, Thoroughbred family 1-x, DEC operating systems, Floors, Canadian Radio Broadcasting Commission, George Albert Smith (film pioneer), DEC microprocessors, Male musical duos, CBC Radio One stations
3	Films featuring an item number, Demoscene software, Yachts, Hardly Art artists, PhyreEngine games, Folk rock, Paris Métro rolling stock, Belgian architecture, Auto races, Naval battles involving the Knights Hospitaller
4	Italian post-rock groups, Grenadian female sprinters, Cable television, MacOS software, Pinnotheroidea, Protein toxins, Austrian record labels, Fininvest S.p.A., Hairstyles, EMC Corporation
5	Eastern Orthodox Christians, Coffeehouses, Authorship debates, Density functional theory software, Korean non-fiction books, Open hardware organizations and companies, American psychedelic rock music groups, Computational chemistry software, Basketball media, Multi-volume biographies
6	Clinics, Fictional golfers, Surf culture, General practice, Lithuanian society, Fictional bakers, Sixth Doctor audio plays, Tops (clothing), Seinfeld, Religion and violence
7	Storm chasing, Moroccan architecture, Weapon operation, Moorish architecture, Rye-based drinks, Madrassas, Confederate States Army, National Weather Service, Meteorological data and networks, Berber architecture
8	Romanian masculine given names, Extinct comets, Train ferries, Burials at Eternitatea cemetery, Danainae, Chinese designers, Rums, Romanian Freemasons, Suffolk Coastal, Greek people
9	Balloon-borne telescopes, Cosmic microwave background experiments, Physics experiments, Minnesingers, Matador Records singles, Mechanical computers, Braniff, Bondage positions, Wine museums, Counting instruments
10	OCaml software, Free theorem provers, Dependently typed languages, Brassica, Educational math software, 4AD singles, Shopping networks, Anthomyiidae, Ponds, Proof assistants

Tabla A.14: Asignaciones a los *topics* para Denver (corpus reducido), parte 1

Topic	Categorías
11	Braniff, American Motors, Cheerleading organizations, Accidents and incidents involving the McDonnell Douglas DC-10, Lumbee, Workplace violence, IPad, Digital Millennium Copyright Act takedown incidents, British anti–nuclear power activists, Can-Am entrants
12	Athletic shoe brands, Laser image generation, Swimwear manufacturers, Starfleet doctors, Sportswear brands, Malaysian male singer-songwriters, New Orleans Jazz draft picks, Fictional medical personnel, Companies listed on the New York Stock Exchange, Oregon
13	Online food ordering, Wesleyan College alumni, Nail care, Philippine television series based on non-Philippine television series, CollegeHumor people, Sun Yat-sen family, Samma tribes, Scientific documents, Vertebrate tribes, Colloids

Tabla A.15: Asignaciones a los *topics* para Denver (corpus reducido), parte 2

Topic	Categorías
0	Shakespearean tragedies, Viva Entertainment, Cultural economics, White culture, MTV India television series, Equal employment opportunity, Films about sexuality, Albania–Italy relations, Philippine music, Multiculturalism
1	Television syndication, Arctiina, Arranged marriage, Landsberg (district), White wine grape varieties, Early American land companies, Human trafficking, Rock formations, Swimming equipment, Purchased territories
2	The Middle (TV series), Fictional drinking establishments, American college cheerleading squads, Berlin Marathon, Naxos, Swing violinists, Texas A&M University traditions, World Marathon Majors, Electronic music organizations, French jazz guitarists
3	Grant Broadcasters, Petra, Plant morphology, Pacific Lutheran University alumni, Skyways, Paramotors, Parasitology journals, Interior designers, Shia Islamic branches, Television criticism
4	Celebrity fandom, Despicable Me (franchise), Professional golf tours, Powhatan Confederacy, Data erasure software, Fictional species and races, Intestinal infectious diseases, Theme restaurants, Modeling, Waterborne diseases
5	Segeberg, Populated places on Lake Kivu, American Forces Network, Glycyrrhiza, Respiratory system, Phaseolus, Lactobacillaceae, Real numbers, Fontana Records artists, CERN software
6	GMG Radio, Guardian Media Group, Fly system, Philippine web media, Global Radio, Madang languages, American boxing referees, Naval aviation, Western South American coastal fauna, Pyrotechnic initiators
7	Synovial bursae, Beds, Pure Noise Records EPs, Playboy TV shows, Arctic land animals, Braids, Usenet alt.* hierarchy, Bareback bronc riders, Dog types, Playboy lists
8	Mexican novels, Yellow symbols, Anomura, English decathletes, Cleveland Crusaders draft picks, Calappoidea, British decathletes, Bridges across the River Trent, National Basketball League (Australia) coaches, A.F.C. Bournemouth
9	Wijchen, Films about radio, Restaurant towers, Ecofeminists, Video game levels, Christian metal, Action Masters, Frequent flyer programs, Transformers lines and sublines, Historic Civil Engineering Landmarks
10	Namibian Afrikaner people, Astrological house systems, Islands District, White Namibian people, Robotics projects, Tail-propeller aircraft, Thripidae, Pain scales, Bhutanese society, Biliary tract disorders

Tabla A.16: Asignaciones a los *topics* para Las Vegas (corpus reducido), parte 1

Topic	Categorías
11	Ukrainian diaspora, Malaysian male singer-songwriters, Dimension, Dutch show jumping riders, Multi-dimensional geometry, Uzbekistani male cyclists, Films about depression, Rabbits and hares, Facial piercings, Ukrainian Canadian culture
12	Dahn, Singaporean literature, Climbing areas, Seafood dishes, Underground hip hop, Saskatchewan provincial electoral districts, Escapements, Uncooked meat dishes, Pain scales, FC Kansas City draft picks
13	ABS-CBN Regional shows, Philippine anthology television series, Washington Wizards executives, Aviation ground support equipment, Eastern Christian liturgy, National Basketball Association team presidents, African-American history between emancipation and the Civil Rights Movement, Proposed buildings and structures, Ship compartments, Telemundo network shows
14	Lithuanian society, Puerto Rican cuisine, Cannabis, PlayStation 2 accessories, Hot sauces, Sound cards, Carolinas Campaign, Belizean society, Climbing equipment, Vehicle wreck ballads

Tabla A.17: Asignaciones a los *topics* para Las Vegas (corpus reducido), parte 2

Topic	Categorías
0	Guatemalan cuisine, Films about emotions, Podemos (Spanish political party), Albanian collaborators with Nazi Germany, Albanian nationalists, Thor Heyerdahl, Pueblo Revival architecture, Films about brothers, Rafts, Fictional motorcycles
1	Citation overkill, Uşak, Works about the Holocaust, Liver anatomy, Grammy Award for Best Contemporary R&B Album, Turkish rugs and carpets, Currency designers, Kayaks, Buxaceae, Albanian cuisine
2	Syntactic transformation, Northwest Los Angeles, Indian whisky, British sportspeople, United Spirits brands, Binghamton University alumni, Transatlantic communications cables, Software bugs, Chinese card games, British sportsmen
3	Tcl programming language family, Software that uses Scintilla, Tsuen Wan, Cleaning, Linux text editors, Sal languages, Cypriot culture, Software using the Mozilla license, Laz people, Dynamically typed programming languages
4	Trent Reznor, Ancient music, Pension funds, Belgian dance music groups, Maya calendars, Anti-ship missiles, Belgian rock music groups, Maya civilization, Batting (cricket), Fictional firearms
5	Commonwealth Games competitors for Guyana, Cable radio, Chinese feminine given names, Music television channels, Courtly love, FM104 presenters, Popping dancers, German female models, Rewriting systems, Arsenurinae
6	Danish death metal musical groups, Marriott International, Marriott International brands, Otis Elevator Company, Norwegian ambient music groups, Utah gubernatorial candidates, Norwegian progressive metal musical groups, Nigerian judges, Danish heavy metal musical groups, Attacks on hotels
7	East German football managers, Korean alcoholic drinks, Eichsfeld (district), Rick and Morty, Sport aircraft, Sorghum, Media events, Central Uplands, Trinidad and Tobago artists, German singer-songwriters
8	Beethoven quadrangle, Bill James, Wayne State University people, Pop instrumentals, British television people, Computational electromagnetics, Private Stock Records singles, Cypress County, Distributed computing problems, Metasongs
9	Ballrooms, Knowledge management, Powerlifting, Galleriini, Natural language processing, Evicted squats, Lawrence Welk, South African poetry, Infoshops, Individual rooms
10	Despicable Me (franchise), Nearctic ecozone fauna, Ok languages, Trans–New Guinea languages, Fictional species and races, Remote desktop software for Linux, Autogyros, Lake Balkhash, Electric power generation, Teaching

Tabla A.18: Asignaciones a los *topics* para Los Ángeles (corpus reducido), parte 1

Topic	Categorías
11	Black British photographers, Gator Bowl, Sheraton hotels, Los Angeles Historic-Cultural Monuments, British photography organisations, Air Force Falcons football bowl games, Starwood Hotels & Resorts brands, Speech perception researchers, American football cornerbacks, Cuneiform
12	Distributors, Korangi Town, Cadet College Hasan Abdal alumni, Individual sailing yachts, Sun workstations, Fictional Australian people, Canon EOS DSLR cameras, Women printers, Geelong West Football Club coaches, Cleaning

Tabla A.19: Asignaciones a los *topics* para Los Ángeles (corpus reducido), parte 2

Topic	Categorías
0	Middle-earth Half-elven, M*A*S*H, Dior, Fictional military medical personnel, Military psychiatry, Bridge design, A1 road (Great Britain), Yugoslav male middle-distance runners, Senegalese people, Japanese vegetables
1	Minnesota Greens, Yoruba-speaking people, Mutineers, Images requiring maintenance, Visual anthropologists, Circuit complexity, New York (state) Liberal Republicans, Peddie School alumni, Cheng Kung-class frigates, British music video directors
2	Clinics, Assassinated Nazis, General practice, Indiana University – Purdue University Indianapolis alumni, Saturday Night Live catchphrases, Treblinka extermination camp personnel, Travel broadcasters, Economic sociology, Sixth Doctor audio plays, Jungle Entertainment artists
3	Software-defined radio, Military radio systems, Lollipops, Kaiser Mountains, Middle-earth realms, Kosciuszko National Park, County Mayo, Sealdah railway division, Jewish cinema, Bletchley Park
4	Cacti, Real-time web, Microblogging, Natural horsemanship, MacOS software, Text messaging, Border ballads, Firefox OS software, WatchOS software, Northumbrian folklore
5	Entrepreneurship organizations, Films about food and drink, Women basketball executives, Defunct British Columbia provincial electoral districts, National Basketball Association team presidents, Films set on islands, Los Angeles Clippers, Australian post-rock groups, Wellington Blaze cricketers, Suspected criminals
6	Female musical duos, Japanese musical duos, Musical duos, Japanese idol groups, Rhopalidae, Brazil–Russia relations, Acer, Musical trios, Brazilian Grand Prix, Songs about Brazil
7	Dance video games, SISMI, Am stars, Malaysian fashion designers, Hip hop books, Exergames, Miocene pinnipeds, Microphones, Canis Major, Pressurized water reactors
8	Adidas brands, Plastic surgery, Dutch confectionery, Waffles, Sydney International, Fictional castles and fortresses, Porphyrias, Fictional pianists, Kings, Major League Soccer
9	Machetes, Venezuelan diaspora, Ptolemaic princesses, Swords, Parasitic acari, Lensman series, Impact craters on Venus, Madonna (entertainer) concert tours, American radio soap operas, Acaridae
10	Theatres, IMAX, Inonotus, France–Israel relations, Film formats, Cavalry tanks, Jamaican rappers, Fijian emigrants to Australia, Maritime flags, Ridges

Tabla A.20: Asignaciones a los *topics* para Nueva York (corpus reducido), parte 1

Topic	Categorías
11	Warner Bros. Nashville singles, British classical music radio programmes, Billboard Canada Country number-one singles, Lenny Kravitz, School dances, Billboard Country Airplay number-one singles, L-type asteroids (SMASS), Jenna Jameson, Welsh poetry, International information technology consulting firms
12	Ford Taurus, Coloured South African people, Theatre magazines, Project Mercury, Diplomats, People educated at Scarborough College, British jazz clarinetists, Endangered Sino-Tibetan languages, Zimbabwean women cricketers, Danish DJs
13	Caribbean music genres, European social liberals, HBO Family, Films featuring Beans (Looney Tunes), Gonzo pornography, Supernatural, Kwaito artists, Functionalist architects, BBC television documentaries about science, Railway buildings and structures
14	Japanese values, Eastern Ghats, Orlando nightclub shooting, Gustavo A. Madero, Samarra, Punk rock venues, Korean food preparation utensils, Japanese society, Citrus sodas, Danish voice actresses
15	Native American health, Cultures, Language software, Veterinary parasitology, Race and health, Political realism, Finnish academics, Routing software, Panthera hybrids, Anarcho-primitivism
16	Dutch heptathletes, Japanese soups and stews, Japanese noodles, Instant noodle brands, Dutch female bobsledders, Atlantic Records, Policy and political reactions to the Eurozone crisis, Al Shamal, United States Fish and Wildlife Service personnel, Tennessee Walking Horses
17	Weighing instruments, Religious autobiographies, Industrial buildings, Design history, American cabaret performers, Bad Religion, Roman gentes, Indian autobiographies, American music historians, Amaranthus
18	Vegetarian organizations, Vegetarian festivals, Onion-based foods, Ballet styles, Lighthouses, Parades, Allium, Argonne National Laboratory, Birds and humans, Onions

Tabla A.21: Asignaciones a los *topics* para Nueva York (corpus reducido), parte 2

Topic	Categorías
0	Colgate Rochester Crozer Divinity School faculty, McKay family, ASME Medals, Appalachian State University faculty, Monochrome video game consoles, Colby College faculty, Weber State University faculty, Ancient Church Orders, Individual wolves, Motorcycle suspension technology
1	Federated Malay States people, Engaged Buddhists, Jaguars, Vietnamese religious leaders, Packard vehicles, Recruitment, Defiance (TV series), West Virginia Tech Golden Bears football coaches, Longwood University, Breguet aircraft
2	Business incubators, Syntactic categories, Hunan University alumni, New Zealand philatelists, Salmon dishes, Uzbekistani male cyclists, Sound trademarks, Kill Rock Stars, Hong Kong racehorse owners and breeders, Religion and violence
3	Pies, Napier aircraft engines, RCA Victor singles, Windows software, Books about the United Kingdom, Flat engines, Peer-to-peer charities, Proprietary software, Employee relations, Basidiomycota
4	Jihadism, Windows software, Salafi movement, Proprietary software, Luxottica, Kennywood, Popular culture, IOS software, Illusive Sounds singles, Cultural appropriation
5	Online real estate companies, Enewetak Atoll nuclear explosive tests, Cotillion Records singles, Turntables, Sermersooq, Nintendo chips, 679 Artists singles, Kayaks, Mathematical cognition researchers, Lemon dishes
6	Scouting ideals, Canadian comedy radio programs, Salutes, Victory Bowl, Computer hardware, International Harvester vehicles, Gladiatorial combat, Indian rock music groups, Fictional disc jockeys, Variety shows
7	Kitchen knife brands, Illinois River, Heritage railways, Cisco products, Illinois Confederacy, Saskatchewan provincial electoral districts, Data recovery, Camcorders, Higher-speed rail, Vaudeville tropes
8	Windows software, Ministers for Defence (Ireland), Fictional Apache people, Proprietary software, People educated at Castleknock College, Calton Hill, Sandia National Laboratories, IOS software, Animal organizations, Unfinished buildings and structures
9	Pennsylvania Railroad locomotives, Quick-Step Floors, Centrarchidae, Fishing television series, Finnish male alpine skiers, Vice offices, Bass (sound), Python-scripted video games, Gordon College (Massachusetts) alumni, Austrian épée fencers
10	Organizational studies, Wiccan priestesses, Heritage railways, Jewellery designers, Formalism (deductive), Punches (combat), Digital organisms, A Nightmare on Elm Street series music, TinyBuild games, Lutherie

Tabla A.22: Asignaciones a los *topics* para Phoenix (corpus reducido), parte 1

Topic	Categorías
11	Indigenous Australian Paralympians, North Palatinate, Shift-and-add algorithms, Cali, Sport climbing, Opisthoproctidae, Digit-by-digit algorithms, Science fiction games, Locomotive boilers, Climbing
12	Canadian black metal musical groups, Planes, Pop Idol contestants, Thioamides, Southeast Asian swords, Vasopressin receptor agonists, Machetes, Ribbon symbolism, Songs about Liverpool, .380 ACP firearms
13	Observances about science, Astronomy education events, Seafood restaurants, Nigerian television actresses, Astronomy events, Anomura, XM Satellite Radio channels, AGCO, Image hosting, Sirius Satellite Radio channels
14	FMA aircraft, Texas A&M Aggies athletic directors, Astro television channels, Windows software, Roman siege engines, Back Lot Music soundtracks, Amara (genus), Proprietary software, Siege engines, Montreal Royals managers
15	Dominican Republic sportspeople, Daugavpils Municipality, Cancelled Commodore 64 games, American runners, New Zealand bass guitarists, Georgian Revival architecture, Australian comics titles, People educated at Saint Kentigern College, Phi Delta Theta, Paraguayan artists
16	Popcorn brands, Non-fiction books about acting, African-American slang, Food storage, Lobbying, Arab-American organizations, New Zealand slang, Jewish cinema, Turntable video games, Hunting

Tabla A.23: Asignaciones a los *topics* para Phoenix (corpus reducido), parte 2

Topic	Categorías
0	Widescreen comics, Short comics, Wildstorm Publications titles, Welsh lawyers, Fantastic art, Cold War policies, Fictional intelligence agencies, Memory disorders, Arborists, Solids
1	Free QDA software, Perry Ellis International brands, Blue cheeses, Peraceae, QDA software, Italian cheeses, Multiplayer hotseat games, Free R (programming language) software, Big Beat Records (American record label) artists, Brazilian theatre directors
2	Populated places on the Thames River (Connecticut), Rome R. IX Pigna, Georgia College & State University, Castniidae, Heat waves, Legal manuscripts, Confucian education, Social issues, J. C. Penney, Government recruitment
3	Sorting offices, Kru languages, Nigerian media personalities, Former post office buildings, Object relations theory, Transformers factions, Ibadan Polytechnic alumni, Prefixes, Fascist rulers, Hong Kong drama television series
4	Netherlandish art, Extracellular matrix proteins, Art genres, Cowparades, XMPP clients, Installation software, Muskingum Fighting Muskies football coaches, Painted statue public art, Software that uses Scintilla, Facebook software
5	Australian urban planners, Muisca Confederation, Personal ordinariates, Norwegian prisoners sentenced to death, High-speed rail, Muysccubun, Women television personalities, Irish law, Thoroughbred family 9-e, Buns
6	Nederpop, Dutch progressive rock groups, Puerto Rican law enforcement personnel, Apple cultivars, United States Nuremberg Military Tribunals, Dutch hard rock musical groups, Battles involving the Qajar dynasty, Postal markings, Psychedelic rock music groups, Storm chasing
7	Toro, Youth conferences, Nairobi, Hmong culture, Iberian art, Pete Waterman Entertainment singles, Frame lamellophones, Dutch sportswomen, Puerto Rican United States Marines, French mass media owners
8	Canoeing, Paddling, Surfing, Boardsports, Nigerian cricketers, English music publishers, BitTorrent clients, Olympic gold medalists for Indonesia, Lightning, Silurian arthropods
9	Thrash metal, Happiness, Computer keyboard models, People educated at Oslo Waldorf School, Ivorian academics, French words and phrases, IBM products, Extreme metal, Hormel, Geri Halliwell
10	Celebrity fandom, Powhatan Confederacy, Gaelic football referees, Mathematical examples, Modeling, Blaxploitation film directors, Dutch Protestants, War casualties, Easy listening music, Clothing

Tabla A.24: Asignaciones a los *topics* para San Francisco (corpus reducido), parte 1

Topic	Categorías
11	Individual killer whales, Diners, Vietnamese soups, Somali Civil War, SeaWorld Parks & Entertainment, Vietnamese noodles, Nursing, Software projects, Great Eastern Railway people, Latin grammar
12	Lutheran universities and colleges, Video game accessories, Yukon Quest, ZeniMax Media, HTC Vive games, Philatelic museums, Lotus Seven replicas, New York (state) Liberal Republicans, Log houses, Virtual reality games

Tabla A.25: Asignaciones a los *topics* para San Francisco (corpus reducido), parte 2

Topic	Categorías
0	Buffet restaurants, Des Moines metropolitan area, Educational robots, Firefighters, Olividae, Superman storylines, Germanic heroic legends, Restaurant franchises, Revenue services, Heroes (TV series)
1	Golf course architects, Guides, Green chemistry, American male golfers, Officials under Yuan Shao, Films about filmmaking, Guiding Light, Statistical analysis, PGA Tour golfers, Generals under Yuan Shao
2	Israeli medievalists, UNC Greensboro Spartans, Women psychologists, Dropsie College alumni, Dark Shadows audio plays, Judaic scholars, The Three Worlds novels, English bishops, Shin Kibayashi, Polish centenarians
3	Delhi Public School Society, Superacids, Tonality, Razor & Tie singles, Accompaniment, American telethons, Sulfur oxoacids, Bass (sound), Tsunami, Ericsson
4	Agile software development, Dancing with the Stars (U.S. TV series), Software development process, Information technology management
5	Cross-platform mobile software, British film critics, Parasitology journals, Singapore Chinese dramas, Database index techniques, Abandoned animals, Level crossing accidents, Geometric data structures, Fictional marsupials, Television New Zealand programmes
6	Rhodes University, LGBT mayors, Whig (British political party), Florida pioneers, Canadian soul music groups, French agnostics, Baltimore and Ohio locomotives, Whiggism, Stabbing survivors, Hamilton family
7	Hampstead, Fictional motorcycles, Nigerian hip hop DJs, Transit-oriented developments, Star Magic, Media about cakes, Nigerian DJs, Australian frontier wars, Filipino DJs, Films about the Internet
8	Sky regions, Scots property law, College football video games, Stellar streams, College football kickoff games, Dutch sportswomen, Oakland Raiders, NHL Winter Classic, NCAA video games, Commerce raiders
9	Nigerian Civil War, California Greens, Anti-globalization activists, East Africa Protectorate, Secretly Canadian EPs, American atheists, American spoken word artists, Jewellery designers, Jewish Ugandan history, Shining (series)
10	Eclipse-class cruisers, Skin care, Engine problems, Guyanese artists, Zeiformes, Scrubbers, Basque Jews, Wet subs, Yarraville Football Club coaches, Phoenix Suns head coaches

Tabla A.26: Asignaciones a los *topics* para Washington (corpus reducido), parte 1

Topic	Categorías
11	Erotic events, Fictional undertakers, The Three Stooges film remakes, Phouvong District, Adoption forms and related practices, Persons involved with death and dying, Funk genres, Object (computer science), Indiana culture, Object-oriented database management systems
12	Acid house groups, Dixieland revivalist clarinetists, Tromsø IL managers, Environmental artists, American rally drivers, Hawthorn Football Club coaches, Norwegian zoologists, Appalachian old-time fiddlers, Luther Seminary alumni, Australian jazz guitarists
13	Free QDA software, QDA software, Free R (programming language) software, Battles involving Hesse-Kassel, Montclair State Red Hawks football coaches, Seattle Storm, 306 Records artists, Anti-Aircraft Command officers, German Jewish theologians, Medical classification
14	Australian fashion, Pakistani music television series, Raw foodism, British fashion, Health ministers, Canadian fashion, Cavalry tanks, Triple Crown Records EPs, Young adult, American fashion
15	Oss, Yazidi, Fictional undertakers, Nineveh Governorate, Persons involved with death and dying, 1. FC Magdeburg managers, Vivendi subsidiaries, Professional wrestling comics, Court physicians, German progressive metal musical groups
16	Monotypic Eurotiomycetes genera, Hip hop phrases, Mexican sportswomen, BitTorrent clients, People with acquired Spanish citizenship, Silurian arthropods, United States Military Academy, Indoor Football League coaches, French male ski jumpers, Prehistoric scorpions
17	EC 3.6.5, Canadian classical violists, Filipino female swimmers, Hungarian television actresses, British women judges, McNeese State Cowboys football, Fox Business Network, Canadian engineers, Hungarian stage actresses, Corporate executives

Tabla A.27: Asignaciones a los *topics* para Washington (corpus reducido), parte 2

A.2.2. Corpus completo

Topic	Categorías
0	Legoland, Miniature parks, BAE Systems facilities, Roman Empire sculptures, Blackstone Group companies, British Aerospace aircraft, Amusement park companies, Eritrean culture, Dallas Mavericks, Ceremonies
1	Margarine brands, Actuarial associations, Military discipline, Geneva College alumni, Mathematical science occupations, Greetings, Liberty Bowl, Transport and the environment, Statistical data types, Natural Resources Defense Council people
2	French Riviera, Fashion occupations, Personal care and service occupations, American hairdressers, Globus (music), Gender and religion, Geopolitical corridors, Marvel Comics Deviants, Sapindales families, Tremors (franchise)
3	Advertising occupations, Hugo Awards, Communication design, Non-fiction books about acting, Tracked armoured recovery vehicles, Puppis, Journalism occupations, LGBT politics, Belgian architecture, Lou Reed
4	Dray Prescot series, Irony, Christmas short stories, Fictional police commissioners, Defunct Manitoba federal electoral districts, Campuses, Nick Fury, Pop punk group discographies, Bridges over the Ottawa River, Jamaican record labels
5	Latter Day Saint universities and colleges, Piae Cantiones, Transformers Mini Vehicles, Hemingway Foundation/PEN Award-winning works, Outback Bowl, Portrait Records singles, Angelo State University, East Baltimore, Soho Square, Southeast Baltimore
6	Oilers Entertainment Group, Stickers, Lubrication, Mega-City One Chief Judges, North Alabama Lions, Pacific Division (NHL), Ecuadorian sculptors, Alaska Baseball League, Romanian Television, Toll-like receptors
7	American consulting businesspeople, Buzz!, Alvis vehicles, X-type asteroids, Thoroughbred family 5, Italian-American cuisine, Hawaii Five-O, Silurian arthropods, Gonzaga art collection, Belgian television series
8	Manchester Cricket Club cricketers, Appaloosa Interactive games, All-England Eleven cricketers, Industrial automation software, Competitive eating, Interstate 73, Belgian heavy metal musical groups, Sephardi Jewish cuisine, Ignition systems, ZTT Records artists
9	Problem behavior, American snack foods, Books about Europe, TsKIB SOO products, Hostess Brands brands, Lake Mead, Nathan Hale, Eurabia, Telemedicine, Hanila Parish
10	Philippine variety television shows, Sports television, Bauxite mines, Acadia University, Fresno State Bulldogs football, Male dancers, Thrushes, Japanese female biathletes, ABS-CBN shows, Action (philosophy)

Tabla A.28: Asignaciones a los *topics* para Chicago, parte 1

Topic	Categorías
11	The Texas Chainsaw Massacre, Defunct Manitoba federal electoral districts, Henderson State Reddies football coaches, Colombian architecture, TransCanada Corporation, Inertial confinement fusion research lasers, Jacksonville State Gamecocks football coaches, Lake Athabasca, Colombian diaspora, Light bombers
12	Judas Priest concert tours, Heavy metal festivals, Defunct jazz clubs, VH1 music shows, Software frameworks, Christmas, German V-2 rocket facilities, Buddhist media, Rock music television series, Australian chefs
13	Trailers, Arctostaphylos, National Hockey League labor relations, Road accidents and incidents, Toronto Argonauts personnel, National Wrestling Alliance shows, Appalachian bogs, Sports labor disputes, Bataan, Streamliners
14	Windows device names, OS/2 device names, DOS device names, DOS drivers, Amalgam Comics supervillains, Nuristani languages, Burr family, Chattanooga metropolitan area, Jerry Garcia, Tennessee

Tabla A.29: Asignaciones a los *topics* para Chicago, parte 2

Topic	Categorías
0	Maryland cuisine, Anacostia River, Experimental cat breeds, Seismic faults, Trans Canada Trail, Italian soups, Vessels, Crab dishes, Nigerian filmmakers, Senja
1	Erotic photography, McKinley Senior High School alumni, Napier aircraft engines, Social events, American prisoners and detainees, Burlesque, Flat engines, Paris Métro line 5, Cabaret, Gangsta rappers
2	Doughnuts, Fictional ogres, Tournament systems, Complexity classes, Chinese fashion designers, Rail fastening systems, Coffee, Tibetan cuisine, Hoses, Kama basin
3	Jennifer Lopez perfumes, Perfumery, Cambodian anti-communists, Free parties, Taylor Swift, Phoenix Suns executives, Moscow Victory Day Parades, Celebrity perfumes, Executed Cambodian people, Bolivarian Missions
4	Films based on Hungarian novels, White Fathers missions, Office and administrative support occupations, Sancti Spíritus, Transition to the New Order, American fiction, Tamil theatre, San Francisco Bay Area freeways, Zazas, Berlin Marathon
5	Australian radio programs, Bible, Heavy machine guns, Toy figurines, Trial and research firearms, Burlesque, Los Angeles Lakers, Contract bridge bidding, Being Inc. singles, Erotic dance
6	Swiss international schools, European racehorses, Peniarth collection, Doughnuts, Adipates, Alarms, Thoroughbred family 4-n, Baseball, Coffee, Rockhampton Region
7	Banking terms, Virtual economies, Texan cuisine, Payroll, British political media, Major League Baseball team mascots, Dutch inventions, E-commerce, New Mexican cuisine, Molson Coors Brewing Company
8	Positive mental attitude, Zen studies books, Three-player card games, Raw foodism, Hopeless Records EPs, World War II raids, Uptown Records singles, English male single skaters, Altice USA, Federal Assembly (Switzerland)
9	Modulidae, Psychedelic rock record labels, Elaphidion, Italian liqueurs, Fictional birds, Mythological substances, Lost and extinct musical instruments, Pulp fiction, Staphylinidae, Shooters (drinks)
10	Structural engineering standards, The Travelers Companies, Israeli chief executives, Pedestrian safety, Inventory, Auto tuning companies, Medical specialties, Oregon, Asset lists, Indiana

Tabla A.30: Asignaciones a los *topics* para Dallas, parte 1

Topic	Categorías
11	Indonesian martial artists, Indonesian rock music groups, VIA Technologies, Alternative rock groups, Routing software, South Korean comedy-drama television series, New Zealand apples, Indonesian pop music groups, IPv6 transition technologies, Washington Wizards head coaches
12	Universal Music Latin Entertainment artists, Penske, Jalisco, Variety (magazine) people, Association football central defenders, Honda Formula One cars, Vegetable soups, Chairs, Auto racing crew chiefs, Flow regimes
13	Christian Identity, Sibling, Vietnam War POW/MIA issues, Rover vehicles, Pro Bowl, African-American cultural history, Middlesex women cricketers, Central Division (NHL), Converts to Mormonism, Fictional tricksters
14	Cattle breeds, Nathan Hale, Single-handed sailing, Agricultural establishments, Dairy cattle breeds, Music podcasts, Radio paging, Tampa Bay Rowdies matches, Thoroughbred family 6-e, Biometrics software
15	Great Midwest Athletic Conference schools, New Zealand cricket coaches, Music journals, Offender profiling, American string quartets, Craft occupations, Compositions for piano quartet, Augustinian orders, New Zealand Test cricketers, Silicon

Tabla A.31: Asignaciones a los *topics* para Dallas, parte 2

Topic	Categorías
0	Inclement weather management, Estée Lauder Companies, Uruguayan stage actresses, Uruguayan female models, Nouakchott, School terminology, Salads, Vietnamese artists, Cosmetics brands, Thai words and phrases
1	Lutherans, Nazarene theologians, English cheeses, Northwest Nazarene University alumni, Denver metropolitan area, Commonwealth Games competitors for Belize, George Fox University alumni, Swedish geneticists, Prefectures, Pacific Gas and Electric Company dams
2	Broadcast call sign disambiguation pages, PBS member stations, Arab sign languages, Memorials to Lal Bahadur Shastri, DEC microprocessors, Cargo aircraft, Floors, Skeena-Queen Charlotte Regional District, DEC operating systems, Thoroughbred family 2-d
3	World War II German radars, Radar warning receivers, HP LaserJet printers, Paris Métro line 5, Aircraft radars, Sheep, Music journalism, The Fugs, Data recovery, Alfred Döblin
4	Military organization, VMware, Ostreidae, Logo programming language family, Hairstyles, Air force ranks, Ishikawa Prefecture, Alvis vehicles, Italian post-rock groups, Oysters
5	Fort Leavenworth, Alcoholic coffee drinks, Eastern Orthodox Christians, Coffee drinks, Open hardware organizations and companies, Korean non-fiction books, Laser awards and associations, Psychiatric specialities, Authorship debates, PowerPC microprocessors
6	Lithuanian society, White American culture, Clinics, Scottish fairy tales, Irony, Syrian nationalism, Surinamese society, Sweaters, Laotian society, Grupo Globo subsidiaries
7	Amateur radio emergency communications organizations, Jayapura, Observation hobbies, Cambodian Buddhist monks, Papua (province) culture, Weapon operation, Medenine Governorate, Storm chasing, Rye-based drinks, Indian wedding traditions
8	Harness racers, Neogene Costa Rica, Disability theatre, Adonis-class schooners, Private railway stations, Guna Yala, Hellenistic religion, Romanian male weightlifters, Dutch heptathletes, Arkham House books
9	Balloon-borne telescopes, Cosmic microwave background experiments, Dallas Mavericks broadcasters, Physics experiments, NBC owned-and-operated television stations, British novels, Lily Allen, Parkland County, Classic television networks, Motorcycle television series
10	Mongol khans, International art awards, Caenorhabditis elegans genes, Cycling events, Chemical element data pages, Dalhem, Umbrella manufacturers, British agriculturalists, Dependently typed languages, Icebergs

Tabla A.32: Asignaciones a los *topics* para Denver, parte 1

Topic	Categorías
11	Penske, Dallas Mavericks broadcasters, FedEx, American college cheerleading squads, Dallas–Fort Worth metroplex, Variety (magazine) people, NBC owned-and-operated television stations, IndyCar Series team owners, Narita International Airport, Texas
12	Frist family, Athletic shoe brands, Kohlberg Kravis Roberts companies, Bain Capital companies, Swimwear manufacturers, Mixed martial artists utilizing judo, Sportswear brands, Companies listed on the New York Stock Exchange, Tennessee, Rutland
13	Chongqing Rail Transit, Lecythis, Gels, Bodyweight exercise, Literature records, Software requirements, Strength training, Jock series, Yellow River, Systems Modeling Language

Tabla A.33: Asignaciones a los *topics* para Denver, parte 2

Topic	Categorías
0	American nursery rhymes, Companies listed on NASDAQ, Early childhood education, Events, MTV Networks Europe, Tactics, Songs about animals, White culture, Nicknames, Telecom Italia Media
1	Modernismo, Jenni Rivera, Cerro Porteño managers, U.S. Città di Palermo, Club Olimpia managers, Paraguayan football managers, Rede Globo telenovelas, Amiiformes, World Food Programme, Brazilian telenovelas
2	Stoves, Companies listed on the Oslo Stock Exchange, Japanese pottery, Mariachi, Chinese restaurants, American fiction, Tex-Mex restaurants, Rhaebo, Millionaires, Barbadian male swimmers
3	Mandalay Resort Group, Catskill High Peaks, Indoor amusement parks, Skyways, Histology, NK1 receptor antagonists, MGM Resorts International, Petra, Staining dyes, Vocal skills
4	Meizu, Celebrity fandom, Japanese-American cuisine, Despicable Me (franchise), Erotic photography, Statistical principles, NASCAR races at Atlanta Motor Speedway, Brand name desserts, Data erasure software, Roots rock music groups
5	Segeberg, Mobile social software, BlackBerry software, Japanese alcoholic drinks, Windows Phone software, Works about alcohol, Web applications, Acid–base disturbances, Vicia, Summits
6	Thoroughbred family 4-k, GMG Radio, Guardian Media Group, Automatic grenade launchers, Global Radio, Oregon police officers, Barbadian cricket umpires, Cycleways, Madang languages, Grindhouse (film)
7	Sports radio, Afrikaans literature, Usenet alt.* hierarchy, Welsh women painters, Deudorix, Warriors FC head coaches, Sledding, Playboy, Shoulder, Sliding vehicles
8	Sicilian Mafiosi sentenced to life imprisonment, Sicilian Mafia Commission, Basque music, Sicilian Mafiosi, Wari culture, Crab dishes, Dean family, Lamar Cardinals and Lady Cardinals, Japanese record labels, Cable ferries
9	Wijchen, Restaurant towers, Pakistan Marines, Pinball, Historic Civil Engineering Landmarks, Slot machines, Ficidae, Films about radio, Prayer, Avar–Byzantine wars
10	Huawei Ascend, Namibian Afrikaner people, Immigrants to Norway, Ecuadorian sculptors, Henderson State Reddies football coaches, Interbasin transfer, Unblack metal musical groups, China Radio International, Chinese criminals, Dutch portrait painters

Tabla A.34: Asignaciones a los *topics* para Las Vegas, parte 1

Topic	Categorías
11	Music based on works, Pile fabrics, Cambodian artists, Fictional hares and rabbits, Lahore Greens cricketers, Lips, Zombie anime and manga, Virgin Records singles, Ukrainian diaspora, Victimology
12	Hardware verification languages, Dutch voice actresses, Escapements, Queensland federal by-elections, Presbyterian synods, Excavating equipment, Rail fastening systems, Delta-opioid agonists, Dahn, The Avengers (TV series)
13	Roller skating, Restaurant staff, Drive-in restaurants, Canal 13 (Chile) shows, Food services occupations, ABS-CBN Regional shows, United States Marine Corps lists, Portuguese-language magazines, Canadian interior designers, Progressive Democrats TDs
14	Aprilia motorcycles, Lithuanian society, Surinamese society, Bluetooth, Cold drinks, Lesotho society, Activity trackers, Cannabis, Colorado Rapids, Cocktails with rum

Tabla A.35: Asignaciones a los *topics* para Las Vegas, parte 2

Topic	Categorías
0	Israeli financial businesspeople, Fonda family, Podemos (Spanish political party), Fontana Records artists, Tiki bars, Miami Hurricanes athletic directors, Fitzrovia, Films about emotions, Spain national football team, Fictional pandas
1	Theosophy, Medieval Russian people, Horn concertos, Turkish rugs and carpets, Liver, Martinican culture, Concertos for multiple instruments, Liver (food), Australian short track speed skaters, Persian rugs and carpets
2	Masses (music), Approximation theory, Indian whisky, Parsing, Approximations, Syntactic transformation, Inupiat people, Death customs, British Columbia Hockey League, Whisky
3	Tcl programming language family, Food and drink appreciation, Perfumery, G proteins, Royal Navy admirals, Cleaning methods, Software that uses Scintilla, Eating disorders, Tsuen Wan, Perfumes
4	Metricated units, Contract research organizations, Formula SAE, Maya art, Maya society, Trent Reznor, Non-SI metric units, European society, Six Flags AstroWorld, Primes (Transformers)
5	Internships, Dutch radio presenters, Japanese dance groups, Graph rewriting, Singaporean female swimmers, Platyzoa, Human resource management, Shea Stadium, Cable radio, Limón Province
6	Propionamides, Anilides, Synthetic opioids, Jamaican sculptors, Marriott International brands, British military officers, Mu-opioid agonists, Danish death metal musical groups, Companies listed on the New York Stock Exchange, Marriott International
7	Vector graphics editors for Linux, Free vector graphics editors, Friday, Pomace brandies, Electrical resistance and conductance, Vogelsbergkreis, Rick and Morty, Sport aircraft, Record Report Pop Rock General number-one singles, Marburg-Biedenkopf
8	Yenisei basin, Beethoven quadrangle, British Poets Laureate, Analytics companies, Pennsylvania State University people, American punk rock groups, Statistical analysis, Compositions for octet, British television people, Compositions for bassoon
9	Knowledge management, Mass, Legalized squats, Ballrooms, Natural language processing, The Pyramid Companies, Western swing, Elimia, NASCAR races at Bristol Motor Speedway, Hang Lung Group
10	Meizu, Project Management Institute, Dimension reduction, Despicable Me (franchise), Embiotocidae, Kernel methods for machine learning, Maturity models, Power Macintosh, DEC microprocessors, Fictional species and races

Tabla A.36: Asignaciones a los *topics* para Los Ángeles, parte 1

Topic	Categorías
11	Baseball equipment, Tooth development, Grapefruit League, Billboard Canada Country number-one singles, Sheraton hotels, Catfish families, Fashion accessories, Electrical breakdown, Nazarene General Superintendents, Buddhists
12	Creative writing programs, Distributors, Yamaha synthesizers, Downtown Los Angeles, California, Imran Khan family, Geographic information systems, Adobe Creative Suite, Burundian male long-distance runners, Los Angeles

Tabla A.37: Asignaciones a los *topics* para Los Ángeles, parte 2

Topic	Categorías
0	Achill Island, Songs about telephone calls, UK R&B Singles Chart number-one singles, Thai language, Dior, Helianthus, Cash Money Records singles, Internet memes, Ambassadors to the Ottoman Empire, Resource extraction occupations
1	American male rappers, Minnesota Greens, Images requiring maintenance, Capitol Records artists, Syrian football managers, Indian Education Service officers, Executive Yuan, Canadian male mixed martial artists, Tougaloo College alumni, Taiwanese diaspora
2	Lihula Parish, Clinics, NorthWestern Corporation dams, Argentine male modern pentathletes, Viljandi County, Syrian nationalism, Italian people with disabilities, Attention, Contemporary art galleries, Grupo Globo subsidiaries
3	Akron Pros coaches, Irish television shows, American male hurdlers, Tinsukia, Kosciuszko National Park, Ohio University, American male sprinters, Southeast Asian traditional medicine, Barracks, Software-defined radio
4	Salsa, Son cubano, LGBT dance, Cacti, Food packaging, United Farm Workers, Dutch male guitarists, Indonesian Islamists, New Zealand television shows, Cesar Chavez
5	LGBT dance, Dutch male guitarists, Ambassadors to Mali, Montferland, Dance events, Mental training, Thoroughbred family 10-a, Mime, Dutch rock guitarists, Theatrical occupations
6	Female musical duos, Cross-dressers, Duke University campus, Japanese musical duos, Australian male professional wrestlers, LGBT dance, Dutch translators, Musical duos, Afro-Latin American culture, Dutch male guitarists
7	Phoenicis Lacus quadrangle, Paris Hilton, Dance and health, Gbaya languages, Ambisonics, Canis Major, Syro-Hittite kings, Angolan people, Majesco Entertainment games, Exercise organizations
8	Hair diseases, Globes, Juniper Networks, Human hair, British art movements, Gingiva, Flushing Meadows–Corona Park, Adidas brands, Waffles, Radiation health effects
9	Machetes, Venezuelan diaspora, Transgender sexuality, Ptolemaic princesses, Free customer relationship management software, Gratis pornography, Agricultural pest mites, Egyptian queens regnant, China–South Korea relations, Lensman series
10	LGBT dance, Dutch male guitarists, IMAX, Israeli fashion designers, Jeep engines, Dance events, R. Kelly, Zeuhl, Ridges, Dutch rock guitarists

Tabla A.38: Asignaciones a los *topics* para Nueva York, parte 1

Topic	Categorías
11	Supermarkets, Music production, Food halls, Legal software, Red Dwarf, Mozilla add-ons, Fifth Harmony, Italian-American cuisine, Isis, Simplexviruses
12	Ford Taurus, Filipino broadcasters, Thoroughbred family 1-d, Zilog microprocessors, West Siang district, Thoroughbred family 19-c, People educated at Scarborough College, Project Mercury, Mercury compounds, Scottish sportswomen
13	Taros, Supernatural (U.S. TV series), European social liberals, Museum collections, Welsh soap opera actresses, Films featuring Beans (Looney Tunes), Barbadian music, DuMont Television Network, Films based on Don Quixote, Welsh Christians
14	Japanese values, Electronic dance music venues, Eastern Ghats, Atlantic Coast Conference schools, Hispanos, Japanese video game producers, Defunct nightclubs, Japanese video game directors, Irish breads, Royalty
15	Native American health, Cat health, Cultures, Kullu, Gypsy punk groups, Crowds, Race and health, Morbilliviruses, Australian male shot putters, Books about crowd psychology
16	Norwegian entertainers, Swedish Grand Prix, Indiana Wesleyan University alumni, Noodle restaurants, Circus owners, Fictional hippopotamuses, Ramen, Snowdonia, Living arrangements, American female archers
17	Weighing instruments, Grammy Award for Best Contemporary Instrumental Album, Government buildings, Fashion occupations, Olei Hagardom, Memories Off, Roman gentes, Signals Intelligence Service cryptographers, Authorship debates, Locks (water transport)
18	Bacterial proteins, Grupo Globo subsidiaries, Iowa Hawkeyes football, Brazilian television series, Vegetarian organizations, Organic farming, The Dresden Dolls, Madison Square Garden Company, Portuguese-language television networks, Onion-based foods

Tabla A.39: Asignaciones a los *topics* para Nueva York, parte 2

Topic	Categorías
0	Colgate Rochester Crozer Divinity School faculty, Holi, Rupnagar, Religious consumer symbols, McKay family, BioArt, Swedish entertainers, ASME Medals, Bandai consoles, Nihang
1	Tea clippers, Jaguars, Software engineering terminology, Danish male modern pentathletes, California clippers, Industrial and organizational psychology, Danish orienteers, Feilding family, Clippers, Upper Centaurus Lupus
2	Companies formerly listed on the New York Stock Exchange, Business incubators, Illinois, Syntactic categories, Startup accelerators, Repairman Jack (series), Lý dynasty, Singaporean sportswomen, Dutch European Commissioners, EC Comics publications
3	Timex Group, Napier aircraft engines, Hum Sitaray, Athletic Bilbao non-playing staff, Scholastic wrestling, Pies, Terry Pratchett, Fictional pianists, Flat engines, Windows software
4	Jihadism, Athletic Bilbao non-playing staff, Anti-Christian sentiment, Windows software, Converts to Sunni Islam, Proprietary software, Terrorism, English Islamists, Targeting (warfare), Benthophilinae
5	Online real estate companies, Commercial logos, New Zealand apples, Nike brands, Taiwanese girl groups, Fictional plumbers, Enewetak Atoll nuclear explosive tests, Storytelling events, Gentianales genera, Tibetan Buddhist monasteries
6	Arizona State Sun Devils, Pac-12 Conference mascots, Electronic Frontier Foundation, Stretched-curd cheeses, Copyright enforcement, Salutes, Carding (fraud), Honda concept vehicles, Living arrangements, Hand gestures
7	Timex Group, Scholastic wrestling, Thoroughbred family 4-g, Graphics, California, Bodybuilding magazines, Illinois Confederacy, Thoroughbred family 1-n, Throwing, Grammy Award for Best Metal Performance
8	Athletic Bilbao non-playing staff, Films based on South African novels, Windows software, Hellcat Records artists, Calton Hill, Proprietary software, Fictional Apache people, American ska musical groups, Pioneer League (baseball) ballparks, Austria–Germany relations
9	Micropterus, Chevrolet engines, Bass family, Human power, Pennsylvania Railroad locomotives, Swedish Muay Thai practitioners, Global natural environment, Fishing video games, Herennii, McLaren racing cars
10	Formalism (philosophy), Punches (combat), Florida, Formalism (deductive), Science fiction, Iranian military aircraft, News, Films featuring Beans (Looney Tunes), Property insurance, Patrickswell hurlers

Tabla A.40: Asignaciones a los *topics* para Phoenix, parte 1

Topic	Categorías
11	Cox Enterprises, Permira companies, American automobile magazines, Sports clubs, Cariban languages, Explosions, United Kingdom tort law, Line printers, Climbing, Science fiction games
12	E-learning, Democratic Party (United States) presidential campaigns, Educational technology companies, Online education, TsKIB SOO products, Hanila Parish, Ribbon symbolism, Demerara-Mahaica, Fingers, Knots and links
13	Morningside College alumni, South Dakota Democrats, Astronomy events, Satellite radio, South Dakota State Senators, American bankers, Astronomy education events, Balboa Peninsula, Public Radio International stations, Observances about science
14	Timex Group, Scholastic wrestling, Athletic Bilbao non-playing staff, Bodybuilding magazines, Victorian Railways railmotors, Kodak cameras, British sausages, Windows software, American football terminology, FMA aircraft
15	Yogurts, Baseball television series, Frozen desserts, HBO network shows, Single-camera television sitcoms, American serial killers, Android (operating system), Bartolomeo Rastrelli buildings, Dominican Republic sportspeople, American sports television series
16	Revlon brands, Food storage, Non-fiction books about acting, Lobbying, Restaurant terminology, Popcorn brands, Homophobic slurs, LGBT politics, Rooms, Tonga

Tabla A.41: Asignaciones a los *topics* para Phoenix, parte 2

Topic	Categorías
0	Widescreen comics, French Sign Language family, Wildstorm Publications titles, Sign languages, Belgian female fencers, Memory disorders, Fictional intelligence agencies, Michael Bloomberg, Impact craters on Venus, Botswana people
1	Blue cheeses, Atlanta Falcons, Comedy tours, Commonwealth Games competitors for the Cook Islands, Perry Ellis International brands, Free QDA software, Italian cheeses, Tibetan literature, Computer occupations, Australian clothing
2	Water goddesses, Rome R. IX Pigna, Cossidae, Paducah micropolitan area, Roman Forum, Stress (linguistics), Populated places on the Thames River (Connecticut), Ancient Megara, Stanford Financial Group, Nymphs
3	United Parcel Service, Sorting offices, Kru languages, Postal infrastructure, Closing ceremonies, Noah, Collision, Logistics companies, Mythimnini, Information systems conferences
4	Netherlandish art, Sony, Cisco protocols, Pertussis, Psychedelic tryptamines, Art genres, Facebook, Electronic paper technology, Installation software, Facebook software
5	Labor studies organizations, Hong Kong breads, Ecuadorian artists, Indian cardinals, Buns, Personal ordinariates, Thoroughbred family 1-n, Ringerike, Villarreal CF B managers, Florida Panthers
6	Street culture, Batak Karo, Saale basin, Postal markings, Painting techniques, Body piercing jewellery, Rock music duos, Writing, Facial piercings, Novels based on Dungeons & Dragons
7	Toro, Chillwave, Latin Grammy Award for Best Female Pop Vocal Album, Polynesian titles, Latin Grammy Award for Best Engineered Album, Toro people, Hawaiian monarchs, Youth conferences, Mosconi Cup, Battles involving Castile
8	Canoeing, Filipino eskrimadors, American stand-up comedy television series, Industrial music services, Cork senior hurling team matches, Marvel Comics vampires, Comedy genres, Bicolano people, Ruff Ryders artists, Tragedies (dramas)
9	Thrash metal, French feminine given names, Computer keyboard types, Happiness, Muscular system, Mexican sportswomen, Computer keyboard models, Pilates, Extreme metal, Odds BK
10	American ska punk musical groups, Third-wave ska groups, Celebrity fandom, Erotic photography, English cuisine, Brand name desserts, Custard desserts, California, Dutch Protestants, Fashion

Tabla A.42: Asignaciones a los *topics* para San Francisco, parte 1

Topic	Categorías
11	Intelligence assessment, Classified information, Individual killer whales, Diners, Somali Civil War, SeaWorld Orlando, National security, Sioux Falls metropolitan area, SeaWorld Parks & Entertainment, Medical testing equipment
12	Minnesota, Westfield Group, Samsung wearable devices, Panorama software, Yukon Quest, Defunct shopping malls, WebGL, Game jam video games, Lambourn Valley Railway, Bazalt products

Tabla A.43: Asignaciones a los *topics* para San Francisco, parte 2

Topic	Categorías
0	Mandarina, Lemons, Woodboring beetles, Mexican electronic musical groups, Diamond aircraft, European Atomic Energy Community, Citron, National Basketball Association history, Hero Honda motorcycles, Baseball occupations
1	Hugo Award for Best Non-Fiction Book winning works, Works about Marilyn Monroe, Feminist essays, Smash (TV series), Essays about literature, Compositions for piano trio, Erotica, Songs about crime, Static program analysis, Guides
2	Israeli medievalists, UNC Greensboro Spartans, Israeli business executives, Fleetwood, Israeli colonels, Georges Simenon, Women psychologists, Stevie Nicks, Israeli women judges, Irish drummers
3	Fantasy sports, Browser-based game websites, Delhi Public School Society, Superacids, Ringsaker, Tonality, Norwegian thrash metal musical groups, Stuart England, Bhagalpur, Ericsson
4	Agile software development, Dancing with the Stars (U.S. TV series), Software development process, Computer programming tools, Ballroom dance, Software design, Information technology management, Australian reality television series, American television series based on British television series, Seven Network shows
5	BBC Light Programmes, Scottish temperance activists, Trinary minor planets, Safavid princes, Level crossing accidents, Veterinary medicine journals, Terms for females, American Zoroastrians, Fox Broadcasting Company executives, Fictional veterinarians
6	Commercial real estate companies, Interrogative words and phrases, Caldera (company), Charleston Renaissance, Ableton Live users, SCO/Linux controversies, U.S. Route 17, American DJs, Delano family, Oxford and Cambridge Universities cricketers
7	Tootsie Roll Industries brands, Scheduled Ancient Monuments, Capricorn Coast, Lollipops, Toy soldier manufacturing companies, Military chaplains, Shipwreck law, Nigerian DJs, Films featuring a Best Supporting Actress Golden Globe-winning performance, Hampstead
8	Stellar streams, Texas Tech Red Raiders, NCAA video games, Northrop Grumman aircraft, Eersel, Mosconi Cup, National Collegiate Athletic Association, National Football League music, Aare drainage basin, Fictional ethnic groups
9	Journalism, Nigerian Civil War, Biafra, Cameroon–Nigeria border, Debates, Shining (series), Science fiction, American human rights activists, Wars involving Nigeria, Debate types
10	French Sign Language family, Personal care and service occupations, Sign languages, Tennessee, Cosmetic surgery, Nemacheilus, Cosmetics, Doridídae, Lawns, Skin care

Tabla A.44: Asignaciones a los *topics* para Washington, parte 1

Topic	Categorías
11	Finnish theatre directors, Fictional undertakers, Phouvong District, American operatic mezzo-sopranos, Motor vehicles, Erotic events, Funk genres, Railway companies, The Three Stooges film remakes, LGBT directors
12	Abilene Christian University alumni, Promise Keepers, American evangelicals, American Protestant missionaries, Texas, American religious leaders, The Texas Chainsaw Massacre, English female shot putters, New Orleans Jazz draft picks, South Australian ministries
13	Quantico (TV series), New Deal agencies, Free QDA software, Humber College faculty, Systemic risk, Hypocorisms, Free R (programming language) software, Belgian musical groups, Gent–Wevelgem, QDA software
14	Broken Bow Records singles, Jeep engines, Raw foodism, Burmese women, MacOS text-related software, Pakistani music television series, World War II raids, Nash vehicles, Sampling controversies, Burmese artists
15	Northern Iroquoian languages, Israeli literary awards, Yazidi, Fictional undertakers, Israeli pacifists, Arata Isozaki buildings, Nineveh Governorate, Native American language revitalization, Religious paramilitary organizations, South African women lawyers
16	Monotypic Eurotiomycetes genera, Gwar, Chilean sportswomen, Comedy rock musical groups, American thrash metal musical groups, Shimmy Disc artists, Piarist Order, Mexican sportswomen, Comedian discographies, Bands with fictional stage personas
17	EC 3.6.5, Austrian inventors, Tunisian artists, Anime and manga critics, Canadian classical violists, Etruscan art, Tunisian painters, Sri Lankan communists, People with acquired American citizenship, Parti Québécois MNAs

Tabla A.45: Asignaciones a los *topics* para Washington, parte 2