ANÁLISIS DE REDES SOCIALES PARA RECURSOS HUMANOS

Fabio Inui **Teresa Martínez** **Javier Quintana** **Silvia Santos**

Versión preliminar December 25, 2017

Dedicamos esta memoria a nuestras familias, sin cuya paciencia sin límites, no hubiera sido posible su elaboración. Agradecemos a nuestros profesores su dedicación y ayuda.

Contenidos

Resumen ejecutivo

Executive summary

Capítulo 1

Planteamiento del proyecto

En este capítulo vamos a describir las ideas y contexto en el que vamos a desarrollar el contenido del proyecto.

1.1 Descripción

Cuando un departamento de Recursos Humanos o una empresa de reclutamiento se enfrenta a una petición para cubrir un puesto vacante o de nueva creación, el proceso suele llevarse a cabo en diversas fases, que podríamos describir del siguiente modo [1]:

1. **Preselección**: etapa inicial en la que se detectan candidatos adecuados para el perfil buscado, bien recurriendo a anuncios en portales especializados, bien con búsquedas personalizadas de perfiles. En esta etapa se elabora una lista de candidatos que pasarán a las siguientes fases del proceso, descartando aquellos cuyas competencias no sean las adecuadas para el puesto.

2. **Entrevista inicial**: en esta etapa los candidatos seleccionados en la etapa anterior son contactados para conseguir ampliar la información de la que se dispone sobre ellos (por ejemplo sobre las aptitudes particulares y experiencias previas consignadas en el CV), y verificar el interés y compromiso del candidato con respecto a la oferta.

3. **Informe**: tras la entrevista inicial, se seleccionan los mejores candidatos para el puesto, y se realiza un informe donde se consignan los datos originales (el CV, por ejemplo) y los datos añadidos en el curso de la entrevista inicial.

4. **Presentación de candidatos**: el empleador recibe el informe elaborado en el punto anterior, y selecciona aquellos que mejor se ajusten a sus necesidades, muy habitualmente realizando nuevas entrevistas con ellos.

5. **Decisión**: es el momento en que se elige el candidato al que se se le va a ofrecer el puesto, etapa en la que puede complementarse la información recogida hasta el momento con referencias recabadas de anteriores empleadores.

6. **Oferta**: etapa en la que la empresa presenta al candidato la oferta en firme, habitualmente por escrito, consignando la voluntad de la empresa de incorporar al candidato y los detalles económicos.

7. : **Seguimiento**: para comprobar que una vez incorporado a la empresa, tanto empleado como empleador están conformes con el resultado del proceso.

Tradicionalmente, el comienzo de este proceso, la detección de candidatos, se realizaba en numerosas ocasiones a través de anuncios en prensa, bases de datos de candidatos construidas a lo largo del tiempo, y la explotación de la red de contactos personales del entorno del empleador. Hoy por hoy, estos métodos tradicionales han sido complementados, y algunos dirían que prácticamente suplantados, por métodos que explotan la información contenida en la web.

Los técnicos de selección se enfrentan a un mundo muy diverso donde tanto la difusión de los posibles puestos como la información sobre los candidatos para los mismos está diseminada en numerosos formatos, teniendo un papel preponderante diversas plataformas o portales web (InfoJobs, Monster, etc.) y redes sociales en general (LinkedIn, Twitter, Facebook, Instagram, etc.). Desde el punto de vista del técnico de selección, las primeras contienen mucha información sobre las aptitudes de los posibles candidatos, sus conocimientos, formación y experiencia, ya que son portales donde los propios usuarios consignan sus curricula vitae, y también sobre su situación laboral actual y expectativas. En el segundo grupo de fuentes, las redes sociales, hay algunas que tienen el carácter específico de las primeras (LinkedIn es el ejemplo más claro), y hay otras en las que se consigna información diversa, llamémoslas de propósito general, tal vez en mayor medida personal que profesional.

El objetivo de nuestro proyecto es complementar el trabajo habitual de un departamento de Recursos Humanos o de un seleccionador de personal en los portales y redes sociales dedicados al mundo laboral, con información laboral extraída de fuentes menos estándar, como son las redes sociales de propósito general. Estas redes son a menudo aprovechadas por los usuarios para difundir mensajes relacionados con su actividad laboral, y una descripción de su actividad en las redes es relevante desde el punto de vista de un reclutador, en la medida que da información del compromiso de la persona con su actividad, su valoración por parte de otros usuarios, su proactividad, etc.

En este trabajo hemos elegido la red social Twitter por diversos motivos: es una red muy dinámica, fácil de usar, rápida y divertida, que involucra cientos de millones de usuarios activos en todo el mundo: 328 millones según la web de la red. El crecimiento del número de usuarios fue casi exponencial entre 2010 y 2014, si bien últimamente la velocidad a la que adquiere nuevos usuarios ha perido intensidad. Es también una red que, desde sus orígenes, ha puesto a disposición de los interesados los mecanismos necesarios para acceder a la información que atesora, con ciertas limitaciones, pero de forma relativamente sencilla.

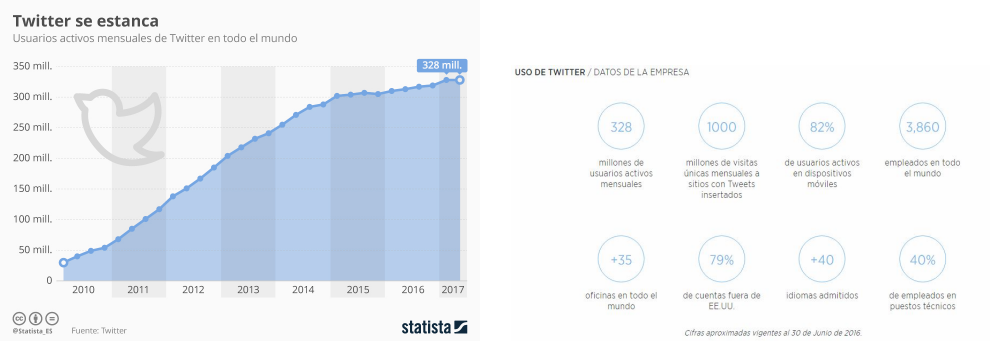


Figura 1.1.1: Twitter: evolución del número de usuarios (Statista 2017, https://es.statista.com/grafico/10476/el-numero-de-usuarios-de-twitter-se-estanca) y datos de la empresa, https://about.twitter.com/es/company.

Esta red da cabida a relaciones diversas, entre usuarios de variada índole. Dado que muchos de los usuarios publican información relacionada con su ocupación laboral, es natural esperar que en Twitter se formen comunidades de individuos que comparten interés en diferentes aspectos de dicho ámbito. Nuestro propósito es definir e implementar un proceso que permita agregar información referente a esas comunidades a un determinado proceso de selección.

Observemos las dos siguientes ofertas de trabajo aparecidas recientemente (Septiembre 2017) en LinkedIn, incluyendo los requisitos solicitados a los posibles candidatos:

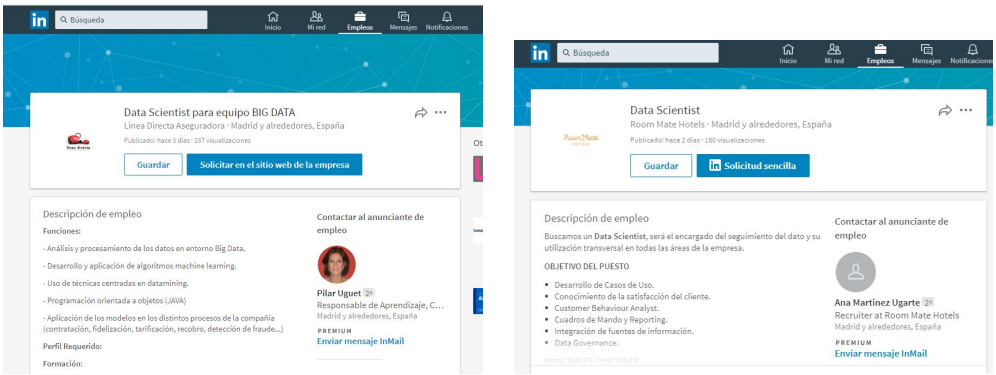


Figura 1.1.2: Dos ofertas de empleo.

Figura 1.1.3: Requisitos de las dos ofertas de empleo.

En ambos casos, entre los requisitos se encuentran conocimientos sobre Python, R, SQL, machine learning y data mining. Un reclutador probablemente usará esas palabras clave para buscar

los perfiles adecuados para alguno de los dos puestos, y construirá un conjunto de posibles candidatos (el primer paso en nuestra descripción del proceso de contratación). En esta fase, y gracias a Octopus Data Insights, nuestro reclutador contará con una ayuda extra. Octopus Data Insights le proporcionará una lista de usuarios de Twitter que hayan publicado contenido en el que aparezcan esas palabras clave, que complementarán el resultado que el reclutador haya obtenido por sus propios medios. La información proporcionada por Octopus Data Insights resultará relevante también más adelante en el proceso, cuando haya que tomar una decisión entre varios candidatos para determinar cuáles son los más adecuados para el puesto: usando la información de Twitter, los usuarios de la lista estarán ordenados según diversos criterios de relevancia.

El proceso para producir la información que ayudará al reclutador en el proceso es el siguiente:

1. Identificar los vocablos que determinan las habilidades que ha de poseer cualquier candidato para la oferta en cuestión y extraer de Twitter aquellos tuits con contenido relacionado con ellos.

2. Dados esos tuits, construir un conjunto de usuarios, que entendemos como posibles candidatos a la oferta.

3. Usar la información publicada por los usuarios para determinar el grado de adecuación a la oferta (serán más adecuados aquellos que hayan publicado información sobre todos los conocimientos requeridos que aquellos que solo hayan publicado sobre alguno de ellos, y más relevantes aquellos más activos, según el número de tuits publicados sobre cada área).

4. Estudiar la relación entre los usuarios de este conjunto, y determinar los más relevantes en sentido relativo (en términos de actividad en la red, cuáles son los más "retuiteados", cuáles los más seguidos, etc.).

1.2 Contexto de negocio

Lo que hay hecho y lo que no.

1.3 Objetivos

Lo que queremos conseguir, qué significa que lo hayamos conseguido.

1.4 Hipótesis y limitaciones

Aquí todo lo que asumamos al plantear el proyecto, y hasta dónde puede llegar. Límites del uso de la información de las redes sociales, límites del proceso en sí (ventana temporal, no detección de todos los candidatos, etc.).

La hipótesis fundamental que estamos haciendo al iniciar este proceso, es que la actividad en Twitter acerca de un determinado tema (por ejemplo, publicar algo relacionado con Python), supone

que el usuario en cuestión tiene conocimientos sobre dicho tema (en nuestro caso, entenderíamos que ese usuario posee conocimientos de Python). Esto es cuestionable, por supuesto, pero también ponderable si tenemos en cuenta que la actividad no sea esporádica. Si un usuario publica sobre un tema en numerosas ocasiones, la hipótesis de que ese tema no le resulta ajeno, va cobrando fuerza.

Entre las limitaciones de las que adolece el proceso definido para llevar a cabo el proyecto, se encuentran las siguientes:

1. En general, no todos los posibles candidatos tienen por qué usar Twitter, y por tanto habrá muchos que queden directamente fuera de nuestro proceso.

2. Los tuits utilizados en el proceso están sujetos a una ventana temporal. Habrá muchos candidatos, usuarios de Twitter, que no aparezcan en nuestros registros, por no presentar actividad durante ese tiempo.

3. Twitter impone limitaciones en la cantidad de información a la que deja acceder, y por ello, también es posible que los usuarios pierdad visibilidad en este proceso, porque el contenido publicado por ellos no se encuentre entre el proporcionado por la red social durante el proceso de extracción de datos.

Otra limitación del proceso es que la información que obtenemos de la red es a nivel de usuario de Twitter. La dirección de correo o el nombre verdadero de la persona en cuestión, o cualquier dato que pudiera identificarla no está necesariamente disponible en la aplicación, salvo que el usuario lo haya querido hacer público explícitamente. Esta información, y la forma en que se utilice, es clave para la usabilidad del resultado del proyecto, en dos aspectos principales:

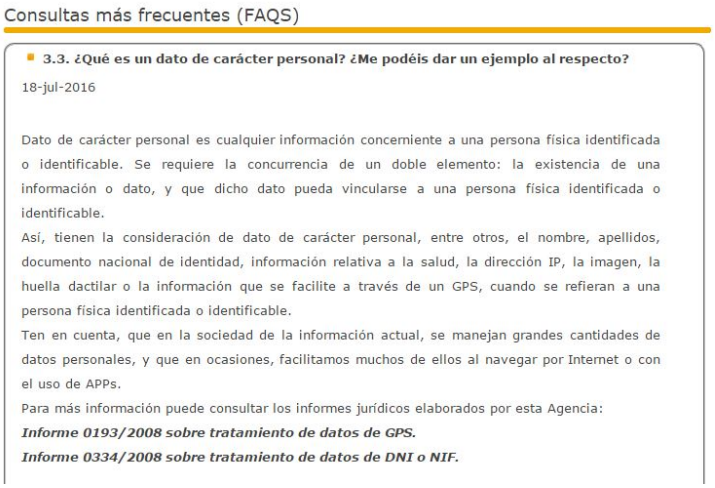
1. para que el reclutador pueda hacer uso de la información, la persona ha de estar identificada, lo suficientemente como para abrir un canal de comunicación entre el reclutador y el posible candidato.

2. Desde el punto de vista de la comercialización del resultado del proyecto, el hecho de identificar usuarios en una red social y usar esa información con fines lucrativos, ha de ser implementado de forma muy cuidadosa. El impacto de la Ley Orgánica de Protección de Datos de Carácter Personal (LOPD) es muy relevante en nuestro proyecto, y merece un apartado especial. Nos ocupamos de ello en la sección 1.4.1.

En relación al primer punto, evidentemente proporcionar un usuario de Twitter ya es abrir un canal de comunicación. Sin embargo, solo la información de las publicaciones del usuario no es suficiente para incluirlo en un proceso de selección, incluso antes del primer contacto entre reclutador y candidato, y el primero probablemente necesitará más información (por ejemplo un CV) para considerar al segundo. Una forma de solventarlo sería cruzar la información de Twitter (el nombre de usuario) con la contenida en otros portales (como LinkedIn, Facebook, Academia.edu, ResearchGate, Glassdoor, etc.), ya que a menudo el usuario de Twitter es parte de los datos consignados en los CV. Esta extensión del proyecto la hemos dejado deliberadamente fuera del planteamiento de este proyecto, aunque sería *conditio sine qua non* para una implementación comercializable del proyecto.

1.4.1 La Ley Orgánica 15/1999, de 13 de diciembre, de Protección de Datos de Carácter Personal

La Agencia Española de Protección de Datos, AEDP, define un dato de carácter personal del siguiente modo:



Consultas más frecuentes (FAQS)

3.3. ¿Qué es un dato de carácter personal? ¿Me podéis dar un ejemplo al respecto?

18-jul-2016

Dato de carácter personal es cualquier información concerniente a una persona física identificada o identificable. Se requiere la concurrencia de un doble elemento: la existencia de una información o dato, y que dicho dato pueda vincularse a una persona física identificada o identificable.

Así, tienen la consideración de dato de carácter personal, entre otros, el nombre, apellidos, documento nacional de identidad, información relativa a la salud, la dirección IP, la imagen, la huella dactilar o la información que se facilite a través de un GPS, cuando se refieran a una persona física identificada o identificable.

Ten en cuenta, que en la sociedad de la información actual, se manejan grandes cantidades de datos personales, y que en ocasiones, facilitamos muchos de ellos al navegar por Internet o con el uso de APPs.

Para más información puede consultar los informes jurídicos elaborados por esta Agencia:

Informe 0193/2008 sobre tratamiento de datos de GPS.

Informe 0334/2008 sobre tratamiento de datos de DNI o NIF.

Figura 1.4.1: AEDP preguntas frecuentes, https://sedeagpd.gob.es/sede-electronica-web/vistas/infoSede/detallePreguntaFAQ.jsf;jsessionid=147951A8A98D206E87F2655B9E96E7EB?idPregunta=FAQ

El perfil en Twitter de un usuario se puede considerar por tanto un dato de carácter personal, en tanto en cuanto permitiría localizar e identificar al usuario, tal vez a través de cruces de la información en Twitter con información adicional (por ejemplo, lo que comentábamos a propósito de encontrar el CV del usuario usando que frecuentemente, el usuario de Twitter es parte de la información contenida en el mismo, y el CV está disponible en otros portales). Sin embargo, en la política de privacidad de Twitter (https://twitter.com/es/privacy) se establece que

> *"Al utilizar cualquiera de nuestros Servicios, usted da su consentimiento para la recopilación, la transferencia, la manipulación, el almacenamiento, la revelación y otros usos de su información según lo descrito en esta Política de Privacidad. Esto incluye cualquier información que elija proporcionar que se considere sensible según la legislación vigente."*

Y también, en relación a la información del perfil o publicaciones:

*"**Información básica de la cuenta**: si opta por crear una cuenta de Twitter, debe promocionar cierta información personal, como su nombre, nombre de usuario, contraseña, dirección de correo electrónico o número de teléfono. En Twitter, su nombre y nombre de usuario siempre se hacen públicos, incluso en su página de perfil y en los resultados de búsqueda, y puede utilizar su nombre real o un seudónimo."*

*"**Tuits, gente que sigue, listas, perfil y otra información pública**: Twitter está principalmente diseñado para ayudarle a compartir información con el mundo. La mayoría de la información que usted nos facilita a través de Twitter es información que nos está pidiendo que hagamos pública. Puede facilitarnos información de perfil para hacerla pública en Twitter, como por ejemplo, una breve biografía, su ubicación, su sitio web, fecha de nacimiento, o una fotografía. Además, su información pública incluye los mensajes que tuitea; los metadatos facilitados con los tuits, tales como cuándo ha tuiteado y la aplicación cliente que utilizó para tuitear; información sobre su cuenta, como el momento de su creación, el idioma, el país y la zona horaria; y las listas que crea, las personas a las que sigue, los tuits que retuitea o marca como Me gusta, y las emisiones de Periscope en las que hace clic o con las que se relaciona de alguna forma (por ejemplo, haciendo comentarios o clic en el icono de corazón) en Twitter. Twitter disemina amplia e instantáneamente su información pública a una amplia gama de usuarios, clientes y servicios, incluyendo motores de búsqueda, desarrolladores y editores que integran contenido de Twitter en sus servicios y organizaciones, tales como universidades, agencias de salud pública y empresas de investigación de mercado que analizan la información en busque de tendencias y conocimiento."*

A tenor de estas afirmaciones, la información que nosotros vamos a manejar en la implementación de este proyecto (nombre de usuario, información del perfil, tuits publicados) es una información de carácter público.

En su enunciado, la LOPD establece que (texto extraído del informe jurídico 2013-0184 de la AEPD http://www.agpd.es/portalwebAGPD/canaldocumentacion/informes_juridicos/otras_cuestiones/common/pdfs/2013-0184_Red-social-y-creaci-oo-n-de-perfiles-de-empleados..pdf)

Establece a este respecto la Ley Orgánica 15/1999 en su artículo 2 que "El régimen de protección de los datos de carácter personal que se establece en la presente Ley Orgánica no será de aplicación: a) A los ficheros mantenidos por personas físicas en el ejercicio de actividades exclusivamente personales o domésticas."

Y define las actividades personales a continuación:

En cuanto a la determinación de que se entiende por actividades personales o domésticas dispone el Reglamento de desarrollo de la LOPD en su artículo 4 que "Sólo se considerarán relacionados con actividades personales o domésticas los tratamientos relativos a las actividades que se inscriben en el marco de la vida privada o familiar de los particulares." Esta es también la interpretación del término "personal" contenida en la Sentencia de la Audiencia Nacional de 15 de junio de 2006 al señalar que "(. . .) Será personal cuando los datos tratados afecten a la esfera más íntima de la persona, a sus relaciones familiares y de amistad y que la finalidad del tratamiento no sea otra que surtir efectos en esos ámbitos.

También dará lugar a la aplicación de la Ley Orgánica 15/1999, por superar el ámbito de la vida privada o familiar de los particulares la publicación de datos de terceros en la red cuando no existan limitaciones de acceso a su perfil, en cuanto que dicha publicación constituye una cesión de datos, definida en el artículo 3 j) de la LOPD como "Toda revelación de datos realizada a una persona distinta del interesado", ya que en estos supuestos, como señalaba la sentencia la Sentencia de 6 de noviembre de 2003 (caso Bodil Lindqvist) del Tribunal de Justicia de las Comunidades Europeas, no se inscribe en el marco de la vida privada o familiar de los particulares "un tratamiento de datos personales consistente en la difusión de dichos datos por Internet de modo que resulten accesibles a un grupo indeterminado de personas."

Refiriéndose incluso a datos de carácter público que aparecen en los datos, la interpretación de la LOPD respecto al uso de los mismos por terceros, especialmente en el caso en el que la finalidad de dicho uso sea comercial, es que el único medio para legitimar el uso es el consentimiento explícito de la persona cuyos datos van a utilizarse. Este consentimiento ha de cumplir unas determinadas condiciones (de nuevo extraemos del informe jurídico 2013-0184 de la AEPD)

El tratamiento de datos de carácter personal debe encontrarse fundado en alguna de las causas legitimadoras previstas en el artículo 6 de la Ley Orgánica 15/1999, disponiendo a este respecto su número primero que "El tratamiento de los datos de carácter personal requerirá el consentimiento inequívoco del afectado, salvo que la ley disponga otra cosa."

(. . .)

Dicho consentimiento debe reunir las características señaladas en el artículo 3.h de la misma Ley que lo define como "manifestación de voluntad, libre, inequívoca, específica e informada, mediante la que el interesado consienta el tratamiento de datos personales que le conciernen".

Esta Agencia ha venido describiendo en sus informes dichas características de manera que se entiende por consentimiento libre aquel que ha sido obtenido sin la intervención de vicio alguno del consentimiento en los términos regulados por el Código Civil.

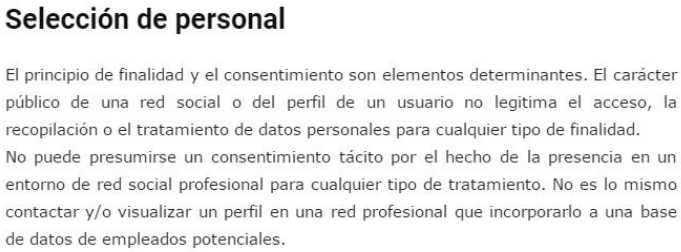


Figura 1.4.2: Ayuda Ley de Protección de Datos, https://ayudaleyprotecciondatos.es/2010/09/16/redes-sociales-empresas-y-proteccion-de-datos/

Como consecuencia de toda esta información, entendemos lo siguiente:

- que los datos del perfil de los usuarios de Twitter, así como sus publicaciones en dicha red, tienen carácter público.
- Que el carácter público de dichos datos no es óbice para poder manipularlos y distribuirlos a terceros, en ninguna actividad que no se circunscriba al ámbito personal o familiar.
- La elaboración de un proyecto de fin de máster no tiene por qué entenderse como una actividad del ámbito personal o familiar, con lo cual no estaría en disposición de difundir esa información a terceros, salvo en las condiciones previstas en la LOPD.
- Cualquier versión comercializable de este proyecto debería contar con los mecanismos adecuados para obtener el consentimiento explícito de los usuarios para el uso de sus perfiles, y posible inclusión en un proceso de selección de personal. Esta fase quedará fuera del plan de elaboración del proyecto.

1.5 Esquema de desarrollo del proyecto

En las siguientes secciones de la memoria iremos estudiando y describiendo las distintas fases que componen el proyecto, que podemos dividir en tres grandes subgrupos:

1. Comenzaremos por hacer un análisis descriptivo de la base de datos recogida a partir del API de Twitter, en relación con el problema que nos ocupa. Estudiaremos la proporción de tuits originales frente a retuiteados, número de tuits descargados por día, número de retuits, relación entre el número de tuits descargados y el número de usuarios distintos, el número de tuits por usuario, y los hastags presentes en los mensajes. Esta tarea la llevaremos a cabo en el capítulo 4.
2. A continuación, comienza la fase de proceso de la información. Nuestro objetivo es constuir una lista de usuarios que podamos recomendar a un profesional de Recursos Humanos como posibles candidatos para una oferta. El primer paso, realizado en el capítulo 5, es entonces seleccionar, a partir de la información descargada de Twitter, los usuarios de interés.

3. La fase final será, con esa lista de usuarios extraídos en la fase anterior, ordenarlos en función de su comportamiento en Twitter. La metodología empleada en este apartado, así como una descripción de los resultados, se abordan en el capítulo 6.

Capítulo 2

Planificación del proyecto

2.1 Equipo

El equipo de Octopus Data Insights está formado por cuatro personas, con perfiles multidisciplinares y complementarios:

- Fabio Inui:
- Teresa Martínez: matemática, con diez años de experiencia en investigación y docencia a nivel universitario, ocho en construcción de modelos de valoración de derivados en empresa financiera de primer nivel, y cuatro de gestión de fondos en una de las principales gestoras españolas.
- Javier Quintana:
- Silvia Santos:

2.2 Desarrollo temporal

Capítulo 3

Infraestructura

En esta sección describiremos la infraestructura que hemos construido para el desarrollo del proyecto.

3.1 Repositorio GIT

El código del proyecto, así como las presentaciones y memoria de este proyecto, está almacenado en el repositorio `https://github.com/MaiteMartinez/MBITProject_Data4all`

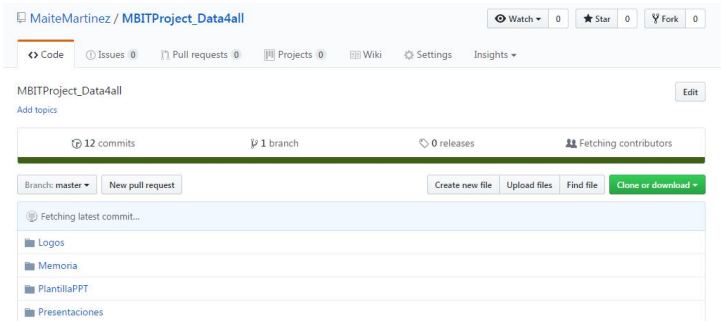


Figura 3.1.1: Repositorio del código del proyecto.

3.2 Infraestructura para la obtención de datos

Como muchas redes sociales, Twitter ofrece acceso a la información que generan sus usuarios a través de un API (*Application Programming Interface*)[7]. El API de Twitter ofrece diversas opciones: Webhook APIs, ADS API, REST APIs y Streaming APIs. La primera está enfocada a generar notificaciones instantáneas a partir de detección de eventos y la segunda a la integración de aplicaciones con la plataforma de publicidad de Twitter. Para nuestro proyecto, solo son relevantes por tanto las dos segundas:

- El API Rest (*Representational State Transfer*) permite realizar consultas puntuales con los parámetros de búsqueda indicados, a través de una componente denominada Search API. El Search API funciona de manera similar, aunque no exactamente igual, a la búsqueda en

la página web de Twitter. El Search API realiza la búsqueda entre una muestra de tuits publicados en los últimos siete días, y las búsquedas están limitadas a 180 peticiones cada ventana temporal de 15 minutos.

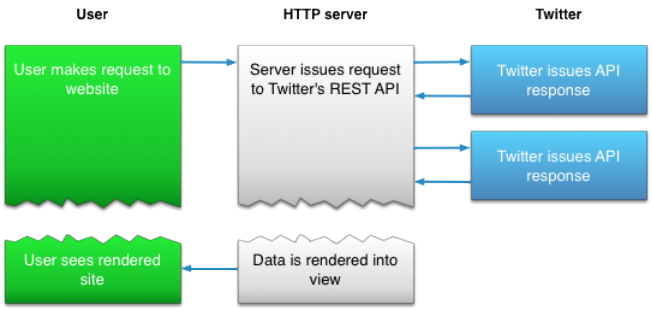


Figura 3.2.1: Funcionamiento del API Rest. https://dev.twitter.com/streaming/overview

La búsqueda realizada por este API está centrada en la relevancia y no en la completitud, lo que quiere decir que algunos tuits y usuarios podrían quedarse fuera.

- El API Streaming permite un acceso con baja latencia al flujo global de tuits de la aplicación, y requiere una conexión HTTP continua. Entre los tipos de flujos disponibles, en la web de Twitter para desarrolladores, se recomienda usar los flujos públicos para realizar minería de datos (en dichos flujos aparecen muestras de los datos públicos de Twitter).

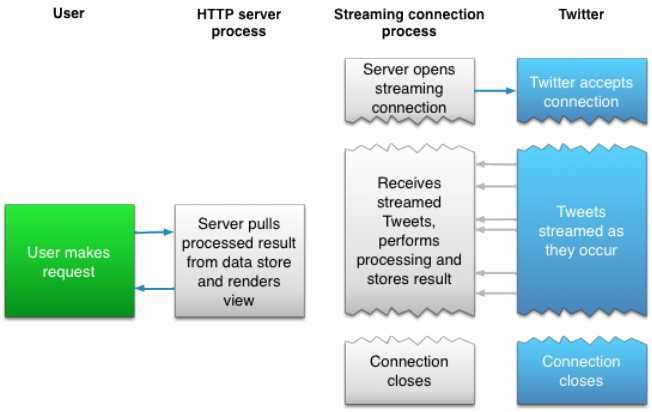


Figura 3.2.2: Funcionamiento del API Streaming. https://dev.twitter.com/streaming/overview

El acceso a ambas versiones de API está gobernado por la autentificación mediante el protocolo OAuth (*Open Authorization*), lo que implica que para cada aplicación deben obtenerse los tokens necesarios de la sección de desarrolladores de Twitter, estableciéndose un número máximo de 7 por usuario. También para ambas versiones del API existen restricciones de acceso. Estas restricciones solo afectan a las versiones gratuitas de los APIs. Hay una versión de pago de este acceso (Twitter

Firehose) que garantiza como respuesta el 100% de los tuits que cumplan los criterios de la búsqueda. Para el desarrollo de este proyecto nos hemos servido del API gratuito, por limitación de costes.

Entre los API REST y Streaming, hemos decidido utilizar el API Streaming, mediante un script en Python usando Tweepy. Este método tiene diversas ventajas y desventajas que pasamos a revisar:

1. El API Streaming proporciona tuits en tiempo real, y funciona como una especie de "grabadora", con la que se van registrando todos los tuits a medida que se van produciendo.

2. Como es un proceso que se mantiene a la escucha y va reflejando entradas según se producen, la infraestructura necesaria para que funcione es algo más complicada que para el API Search. Se necesita, por ejemplo, una conexión contínua a Internet, ya que el tiempo que el proceso no esté corriendo, no estaremos recogiendo tuits.

3. El límite de bajada de los tuits en el API Stream es de 50 tuits por segundo. En nuestro caso, el número de tuits que se producen no pasan de unas decenas por minuto, con lo cual nos aseguramos un barrido bastante exhaustivo de la actividad relevante.

4. Además, al ir bajando tuits consecutivamente, minimizamos el problema de tener tuits repetidos.

5. Como desventaja, no podemos acceder a tuits antiguos, y solo tendremos aquellos que se produzcan en el tiempo que esté el proceso de "escucha" levantado.

Con respecto al segundo punto, hemos bajado los tuits mediante un script de Python, que usa la librería Tweepy, y en el que a medida que van llegando los tuits a un proceso de escucha, los vamos almacenando en una base de datos MongoDB.

```
#Listener
class StreamListener_to_db(StreamListener):
    def __init__(self, collection):
        self.collection = collection
        print("************  New stream created at " + str(datetime.datetime.now()))

    def on_data(self, data):
        tweet_json = json.loads(data)
        try:
            tweetid = self.collection.insert_one(tweet_json).inserted_id
            print ( "========   twit loaded " + str(tweetid) +" at "+str(datetime.datetime.now()) )
            return True # keep stream alive
        except BaseException as e:
            print ('failed ondata,',str(e))
            time.sleep(5) # reload stream


    def on_error(self, status):
        text_error = '************ ERROR ** Status code = %s at %s\n' % (status_code,datetime.datetime.now())
        print (text_error)
        return True # keep stream alive

    def on_exception(self,exception):
        text_error = '************ EXCEPTION  ** %s at %s\n' % (exception,datetime.datetime.now())
        print (text_error)
        return True # keep stream alive

    def on_timeout(self):
        #print 'paso por on_timeout\n'
        text_error = '************ TIMEOUT at %s\n' % ( datetime.datetime.now())
        print (text_error)
        return False #restart streaming
```

Figura 3.2.3: Función de escucha de tuits.

Para mantener vivo el proceso de escucha y que no caiga frente a errores o timeouts, incluimos en el script un control de incidencias:

```
exit = False
while not exit: # Making permanent streaming with exception handling
    try:
        stream = Stream(auth, l)
        stream.filter(track=query,languages = ["es"])
    except KeyboardInterrupt:
        print ('\nGoodbye! ')
        exit = True
    except:
        print ("Error. Restarting Stream....  ")
        time.sleep(5)
```

Figura 3.2.4: Gestión de incidencias en el proceso de escucha.

Y para refrescar el proceso, y que la conexión con Twitter funcione correctamente de forma continua, se programaron relanzamientos diarios del script en el programador de tareas de Windows.

3.3 Lenguajes de implementación

Para el desarrollo de la mayor parte de la lógica del proyecto hemos elegido Python 3.6 (a través de la instalación de Anaconda). Esta elección tiene diversas ventajas, en las varias fases del proyecto, ya que este lenguaje facilita las siguientes tareas:

- Comunicación con el API Streaming de Twitter a través del paquete Tweepy.
- Sencilla interacción con el formato JSON, que es en el que los tuits son descargados desde el API, gracias al paquete json.
- Comunicación con la base de datos documental MongoDB, a través del paquete pymongo.
- Incorporación nativa del encoding UTF-8 ("Unicode Transformation Format" con números de 8 bits), lo que nos ahorra muchos quebraderos de cabeza al tratar con caracteres especiales del español (tildes, ñ, etc.) y la presencia de emojis en los textos de los tuits.[1]

3.4 Almacenamiento y manipulación de los datos

Para la primera toma de contacto con los datos descargados del API de Twitter, nos hemos decantado por almacenarlos según íbamos recogiéndolos en una base de datos MongoDB. Las razones son varias:

- MongoDB trata de forma natural documentos en formato JSON.
- También maneja sin problemas documentos con encoding UTF-8[2], lo que nos ayuda a no tener problemas de encoding al almacenar los tuits.
- Se integra muy bien en programas en Python gracias al paquete pymongo.

[1] https://docs.python.org/3/howto/unicode.html.
[2] https://docs.mongodb.com/v3.4/reference/bson-types/

- No necesita una definición de la estructura del documento, lo que es muy conveniente para tratar tuits, donde el número de campos que obtenemos del API y el contenido de esos campos no siempre sigue el mismo formato. Este punto es en particular relevante si el API de Twitter cambiase, o se introdujeran nuevos campos. Por ejemplo: a partir del 26 de Septiembre de 2017, Twitter cambió el límite de 140 caracteres a un límite de 280 caracteres a algunos usuarios, y eso provocó que aparecieran nuevos campos en el cuerpo de cada tuit[3].
- Es una base de datos bastante rápida, y escalable.

3.5 Desarrollo en la nube

[3] https://developer.twitter.com/en/docs/tweets/tweet-updates

Capítulo 4

Tratamiento inicial de los datos: análisis descriptivo

4.1 Descripción de los datos

En pantalla, un tuit relevante para nuestro proyecto podría ser el mostrado en la figura adyacente. Sin embargo, cuando descargamos el mismo tuit a través del API Search de Twitter, obtenemos mucha más información, en formato JSON.

Figura 4.1.1: Ejemplo de tuit.

El formato JSON (*Javascript Object Notation*)[1] es un formato ligero de intercambio de datos, fácilmente interpretable (por humanos y por máquinas). Es un formato ampliamente utilizado, siendo numerosos los lenguajes que son capaces de usar este formato (lenguajes de la familia C,Javascript, PHP, Python, etc.). En JSON se pueden representar dos tipos de estructuras: un conjunto de pares (clave,valor) con la sintaxis {clave1:valor1, clave2:valor2,...}, también denominado *objeto*, y un conjunto ordenado de valores con la sintaxis [valor1, valor 2,...] (que se denomina *arreglo*). Un valor puede ser una cadena de caracteres con comillas dobles, un número, un valor booleano o nulo, un objeto o un arreglo. Esta flexibilidad permite representar datos de gran complejidad.

Como veremos en el siguiente ejemplo de un tuit en formato JSON descargado a través del API Search de Twitter, puede resultar de ayuda pensar en un JSON como en un diccionario {clave:valor},

[1] http://www.json.org/json-es.html

donde las claves son cadenas de texto, y el valor es algo flexible que acomoda desde una cadena de texto a un vector de objetos o un nuevo diccionario. En particular, la información del tuit que considerábamos más arriba luce de la siguiente manera:


```
{'_id': ObjectId('59e0cbb03842ed08188233d7'),
  'contributors': None,
  'coordinates': None,
  'created_at': 'Wed Oct 11 08:13:41 +0000 2017',
  'entities': {'hashtags': [{'indices': [90, 106],'text': 'MachineLearning'},
                            {'indices': [107, 114],'text': 'Python'}],
               'symbols': [],
               'urls': [{'display_url': 'twitter.com/i/web/status/9...',
                         'expanded_url': 'https://twitter.com/i/web/status/918026805080182785',
                         'indices': [116, 139],
                         'url': 'https://t.co/1WuwNRzn8z'}],
               'user_mentions': []},
  'favorite_count': 1,
  'favorited': False,
  'geo': None,
  'id': 918026805080182785,
  'id_str': '918026805080182785',
  'in_reply_to_screen_name': None,
  'in_reply_to_status_id': None,
  'in_reply_to_status_id_str': None,
  'in_reply_to_user_id': None,
  'in_reply_to_user_id_str': None,
  'is_quote_status': False,
  'lang': 'es',
  'metadata': {'iso_language_code': 'es',
               'result_type': 'recent'},
  'place': None,
  'possibly_sensitive': False,
  'retweet_count': 0,
  'retweeted': False,
  'source': '<a href="http://twitter.com"  rel="nofollow">Twitter Web Client</a>',
  'text': 'Vamos preparando el siguiente libro a estudiar que al final la movida
           me está gustando... #MachineLearning #Python...  https://t.co/1WuwNRzn8z',
  'truncated': True,
  'user': {'contributors_enabled': False,
           'created_at': 'Sat Dec 12 09:13:46 +0000 2009',
```

```
'default_profile': False,
'default_profile_image': False,
'description': 'Me gustan las camisetas y las zapatillas // Desarrollo y '
        'Diseño para sistemas Apple // Creador de @GetPomodoroApp · '
        '@GetAtentoApp · @MADatBUS · @GetMeteo y...',
'entities': {'description': {'urls': []},
             'url': {'urls': [{'display_url': 'desappstre.com',
                               'expanded_url': 'http://desappstre.com',
                               'indices': [0,23],
                               'url': 'https://t.co/oYY42NlHrT'}]}},
'favourites_count': 1512,
'follow_request_sent': False,
'followers_count': 211,
'following': False,
'friends_count': 209,
'geo_enabled': True,
'has_extended_profile': True,
'id': 96309647,
'id_str': '96309647',
'is_translation_enabled': False,
'is_translator': False,
'lang': 'es',
'listed_count': 109,
'location': 'Madrid — Mundo Real™',
'name': 'Adolfo ™',
'notifications': False,
'profile_background_color': '000000',
'profile_background_image_url': 'http://abs.twimg.com/images/themes/theme2/bg.gif',
'profile_background_image_url_https': 'https://abs.twimg.com/images/themes/theme2/bg.gif',
'profile_background_tile': False,
'profile_banner_url': 'https://pbs.twimg.com/profile_banners/96309647/1501577205',
'profile_image_url': 'http://pbs.twimg.com/profile_images/888396793024794624/O6gHh-
lJ_normal.jpg',
'profile_image_url_https': 'https://pbs.twimg.com/profile_images/888396793024794624/O6gHh-
lJ_normal.jpg',
'profile_link_color': '1B95E0',
'profile_sidebar_border_color': '000000',
'profile_sidebar_fill_color': '000000',
'profile_text_color': '000000',
'profile_use_background_image': False,
```

'protected': False,
'screen_name': 'FitoMAD',
'statuses_count': 6893,
'time_zone': 'Madrid',
'translator_type': 'none',
'url': 'https://t.co/oYY42NlHrT',
'utc_offset': 7200,
'verified': False}}

La descripción de cada campo de los que integran el tuit puede encontarse en la página web de Twitter para desarrolladores, https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/tweet-object.

4.2 Obtención de los datos

Los tuits que componen nuestro corpus de datos los hemos obtenido a través del API Streaming de Twitter, a través de una búsqueda dirigida en el API Streaming. Esta búsqueda dirigida se ha realizado a través de palabras clave, asociadas a la actividad de data science, concretamente:

"machine learning"	"machinelearning"	"datamining"	"data mining"	"Python"
"SQL"	"hadoop"	"bigdata"	"big data"	"pentaho"
"rstats"	"SAS"	"tableau"		

Esta petición se define como un vector en Python, donde la coma significa "OR"y el espacio dentro de las comillas significa "AND", y se incluyeron dichos términos con y sin almohadilla (#).

También hemos incluido en la búsqueda un filtro por idioma, incluyendo el parámetro "languages = ["es"]"en la llamada al API, con el objetivo de bajar solo tuits en un idioma. En principio[2] esta búsqueda debería devolver tuits que la aplicación Twitter ha detectado como escritos en idioma español. Sin embargo, también bajamos tuits en otros idiomas (inglés, sobre todo), lo que nos obligará a incluir esta variable en el proceso de selección de usuarios, como veremos en la sección **??**.

El script en el que se realiza la llamada al API de Twitter y el primer almacenamiento de los tuits es el script llamado **download_tweets_stream.py**. Este script importa otros denuestro proyecto, como **OpenMongoDB.py**, que gestiona la conexión a la base de datos MongoDB en el que se almacenarán los tuits. El lanzamiento programado de la tarea se hizo con una entrada en el gestor de Tareas Programadas del portátil, a través del archivo **streaming_upload.bat**. Todos estos archivos se encuentran en el repositorio de GitHub descrito en la sección 3.1.

[2] https://developer.twitter.com/en/docs/tweets/filter-realtime/guides/basic-stream-parameters

4.3 Almacenamiento

Según van produciéndose los tuits, y nuestra "grabadora" los va detectando, los hemos almacenado en una base de datos de MongoDB, en local.

4.4 Revisión inicial de los datos

Una vez almacenados los tuits, realizamos un análisis exploratorio para estudiar con qué material contábamos para el desarrollo del proyecto. Este estudio se lleva a cabo en el archivo **analisis_exploratotio.py** del repositorio de GitHub.

Tuits repetidos

Aunque hemos usado una conexión con el API Streaming de Twitter, que va almacenando los tuits según van publicándose, de los 14.736 documentos que hemos recogido en la base de datos, solo 14.727 son únicos, y hay 9 repetidos. Esto es extraño, pero podría deberse a dos procesos de Streaming corriendo en paralelo sobre la misma base de datos momentáneamente. Como son muy pocos, no los hemos eliminado para hacer este estudio previo de los datos.

El texto de los tuits

Nuestra primera duda es si los tuits que hemos bajado corresponden realmente a contenido relacionado con la ciencia de datos. Esto será objeto de una sección propia a la hora de limpiar los datos. De momento, una forma gráfica de ver si los textos de los tuits se corresponden con el tema que nos interesa es construir una nube de palabras con dichos textos. Aparentemente no hay grandes desviaciones del tema a tratar, ciencia de datos y big data:

Figura 4.4.1: Texto de los tuits

Tuits originales frente a tuits retuiteados

Nos interesaba saber qué proporción de tuits de los que hemos bajado son tuits originales y qué proporción son tuits retuiteados. En general, en los documentos obtenidos a través del API Streaming, Twitter marca con un "RT" al comienzo del texto del tuit aquellos que son retuiteados, a la vez que incluye en el cuerpo del tuit la información completa sobre el tuit que ha sido retuiteado.Hemos observado que hay tuits que parece que son retuits de otros (que comienzan por "RT"), pero luego no tienen los campos "retweeted_status" o "quoted_status". Estos podrían ser mensajes procedentes de bots, que los envían de forma automática. De los 14.736 tuits, hay 6.421 originales, 8.048 retweets estándar y 267 aparentes retuits que no tienen información de lo retuiteado.

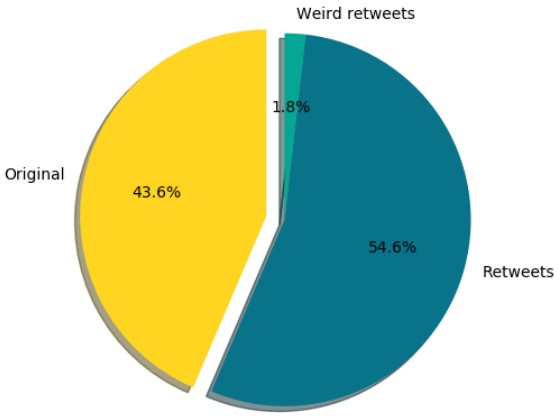


Figura 4.4.2: Proporción de tuits originales y retuiteados.

Número de retuits por tuit

A la vista del gráfico anterior, parece que la actividad principal que hemos captado es una actividad de difusión, en la que los usuarios retuitean información de otros usuarios. Entre estos tuits retuiteados, nos interesa estudiar el número de veces que cada tuit ha sido retuiteado. Nos vamos a fijar en aquellos tuits que aparecen como retuiteados en nuestra muestra, es decir, aquellos tuits cuya información viene en los campos "retweeted_status" o "quoted_status", ya que son indicativos de los temas que interesan a los usuarios. En los 8.048 tuits de nuestro corpus que son retuits con info de lo retuiteado, se han retuiteado 2.686 tuits distintos. La distribución del número de retuits por cada uno de estos 2.686 aparece en las siguientes gráficas.

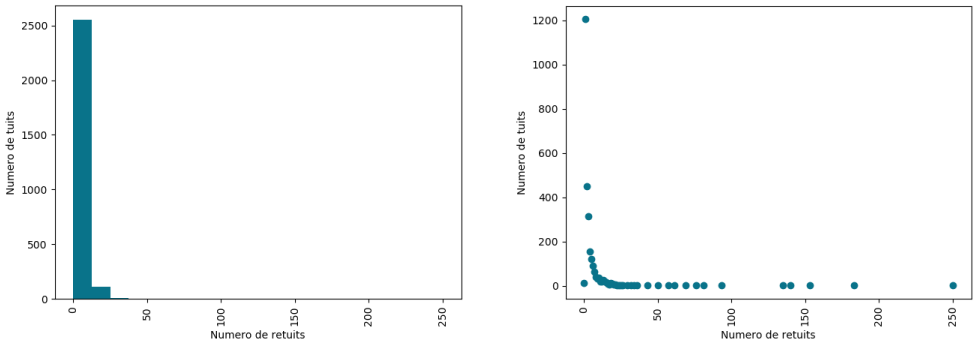


Figura 4.4.3: Retuits: número de retuits por tuit.

Tanto en el histograma con en el gráfico de puntos, es claro que la mayoría de tuits han sido retuiteados unas pocas veces (entre 1 y 5), y que solo unos pocos han sido retuiteados muchas veces. Los dos más tuiteados aparecen en la siguiente figura:

Figura 4.4.4: Retuits: los dos tuits más retuiteados.

Viendo los tuits, parece que podemos encontrar usuarios relevantes para nuestro proyecto no solo en los usuarios que han publicado los tuits que hemos descargado, sino también en los usuarios que han publicado los tuits que los primeros han retuiteado.

Tuits descargados a lo largo del tiempo

Hemos descargado tuits durante dos semanas a finales de Noviembre de 2017. La distribución temporal del número de tuits obtenidos es la siguiente:

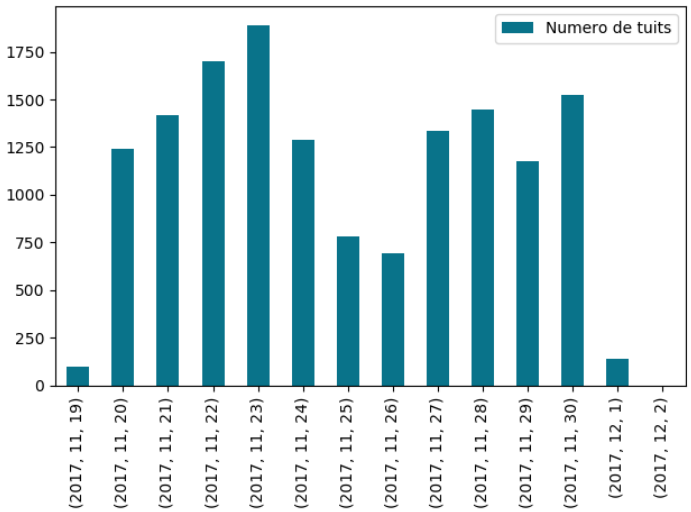


Figura 4.4.5: Tuits descargados diariamente.

El día 19 de Noviembre fue domingo, parece por tanto que se aprecia cierto patrón estacional en la actividad (tuiteándose más entresemana que en fin de semana). Los días 22 y 23 de Noviembre hubo un evento BigData, el V Encuentro de Big Data en Castilla y León, que tal vez tenga relación con la mayor actividad durante esos días.

Relación entre número de tuits descargados y número de usuarios distintos

Para nuestro proyecto, lo más relevante de la base de datos con la que trabajamos es qué información sobre los usuarios podemos extraer a partir de los tuits almacenados. El primer paso suele ser contar, y por ello miramos cuántos usuarios distintos podemos obtener de ella. El siguiente gráfico representa el número de usuarios distintos en función del número de tuits descargados, a lo largo del tiempo.

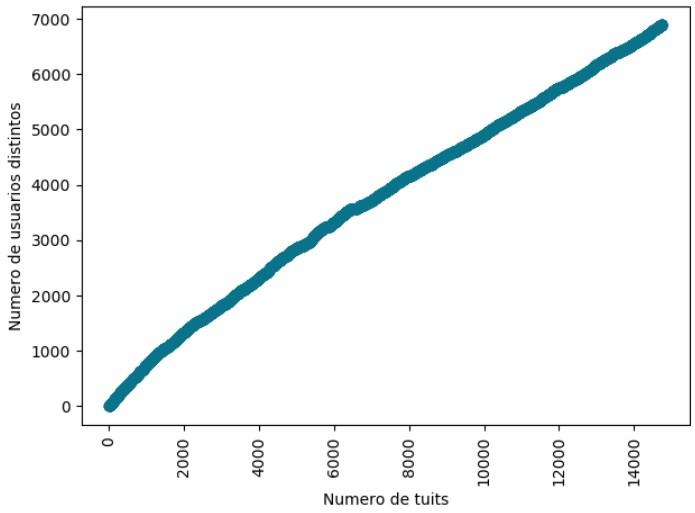


Figura 4.4.6: Número de usuarios distintos frente a número de tuits descargados diariamente.

Este gráfico no presenta sorpresas, y se aprecia que la relación entre el número de tuits descargados y el número de usuarios distintos es prácticamente lineal.

Tenemos 6.890 usuarios distintos (en los usuarios de los tuits descargados, sin contar los usuarios de los tuits que estos usuarios hayan podido retuitear), lo que arroja una media de unos 2.13 tuits por usuario. Ya sabemos que la media es una medida muy poco robusta. Si queremos hacernos una idea de la actividad de nuestros usuarios, mejor estudiamos un poco más la distribución de esa variable.

Número de tuits por usuario

Al igual que hicimos con el número de retuits por tuit retuiteado, veamos un poco más en detalle la distribución del número de tuits que cada usuario ha publicado:

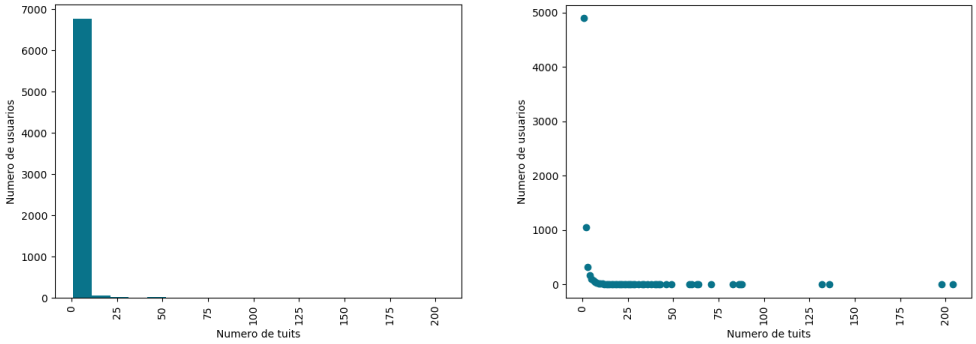


Figura 4.4.7: Número de tuits por usuario.

También en este caso se aprecia que la mayoría de los usuarios han tuiteado una vez, pero que hay algunos usuarios con un número muy elevado de tuits:

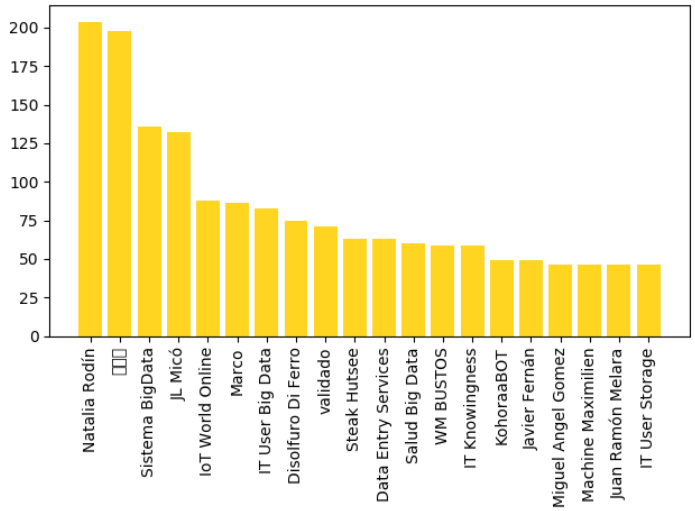


Figura 4.4.8: Número de tuits de los usuarios más activos.

De estos usuarios:

- Natalia Rodín, Sistema BigData, Steak Hutsee parece que ya no existen en Twitter
- IoT World Online es un grupo de personas interesadas en el mundo del Big Data.
- IT User Big Data, validado, Data Entry Services, IT User Storage no está claro si son personas (solo retuits, sin bio o con bio poco descriptiva).
- Salud Big Data es un portal de noticias
- IT Knowingness es un feed
- KohoraaBot, Machine Maximilien son bots

En resumen, solo ocho de los veinte usuarios con mayor actividad parecen adecuados para nuestro proyecto. Esto es indicativo de que la labor de filtrado de la base de datos va a ser clave para conseguir un producto exitoso.

Localización de los tuits

Desde el punto de vista de la comercialización del producto, parece relevante incorporar información geográfica acerca de los candidatos. Para ello, siguiendo el enfoque de este trabajo, deberíamos usar información de este tipo extraída del corpus de tuits descargados de Twitter. Esto nos lleva a preguntarnos cuántos de ellos tienen el dato disponible.

Según la documentación del API de Twitter[3], los tuits pueden asociarse con una localización, generando un tuit que ha sido "geolocalizado", y estas localizaciones pueden ser un punto exacto (con coordinadas de longitud y latitud) o un lugar como una ciudad o país o un área definida mediante una "bounding box". Hay dos campos en un tuit que se usan para describir la geolocalización: "coordinates" y "place". El objeto "place" siempre está presente en el caso de que el tuit esté geolocalizado, mientras que las coordinadas solo en el caso en el que el tuit tenga asignada una localización exacta. En el siguiente gráfico hemos representado el porcentaje de tuits con el campo "place" con valores, en el que podemos apreciar que, salvo que cambiásemos la búsqueda en Twitter, no es un campo que vayamos a poder usar en el análisis:

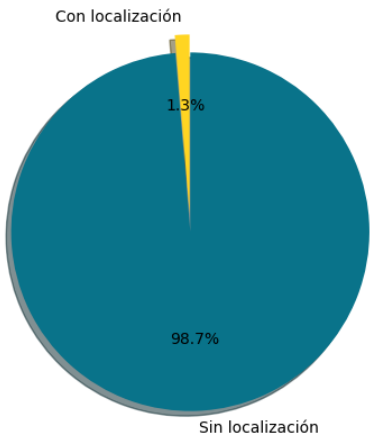


Figura 4.4.9: Proporción de tuits geolocalizados.

Hashtags: distribución

En nuestro proyecto, va a ser muy importante cómo clasifiquemos los contenidos de los tuits. Los propios usuarios clasifican los contenidos de los tuits cuando los etiquetan a través de los denominados "hashtags", que son palabras precedidas por el símbolo # (almohadilla o *hash*).

Hemos estudiado los hashtags incluidos en los tuits de la base de datos, tanto los de los tuits descargados, como de aquellos citados o retuiteados en ellos. Los siguientes gráficos dan una idea de los hashtags presentes en ambos casos:

[3] https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/geo-objects

Figura 4.4.10: Hashtags en los tuits descargados (arriba) y en los tuits descargados y citados y retuiteados en ellos.

En principio parece que no difieren mucho. Vamos a comparar los hashtags más frecuentes en ambos conjuntos de tuits:

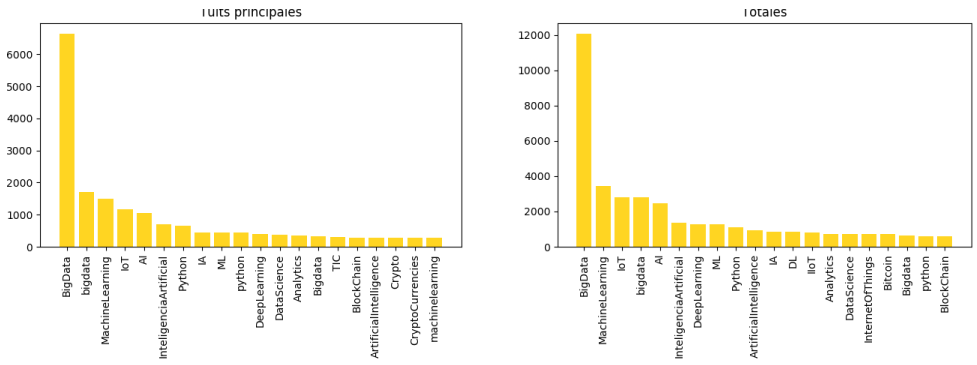


Figura 4.4.11: Hashtags más frecuentes en los tuits descargados (izq.) y en los tuits descargados y citados y retuiteados en ellos.

Perfiles de los usuarios

A la hora de desarrollar el proyecto y construir la lista de usuarios recomendados, es importante filtrar aquellos que sean efectivamente personas, y desechar aquellos que sean empresas, bots, cuentas oficiales de organismos públicos, etc.

Este análisis lo llevaremos a cabo usando la información contenida en el campo "user" de los tuits, y será necesario para ello examinar el texto de la bio que incorpora, campo que registra la información que los propios usuarios facilitan sobre ellos mismos. Para hacernos una idea del tipo de texto y contenidos a los que nos enfrentamos, hemos dibujado también una nube de palabras que represente esta información.

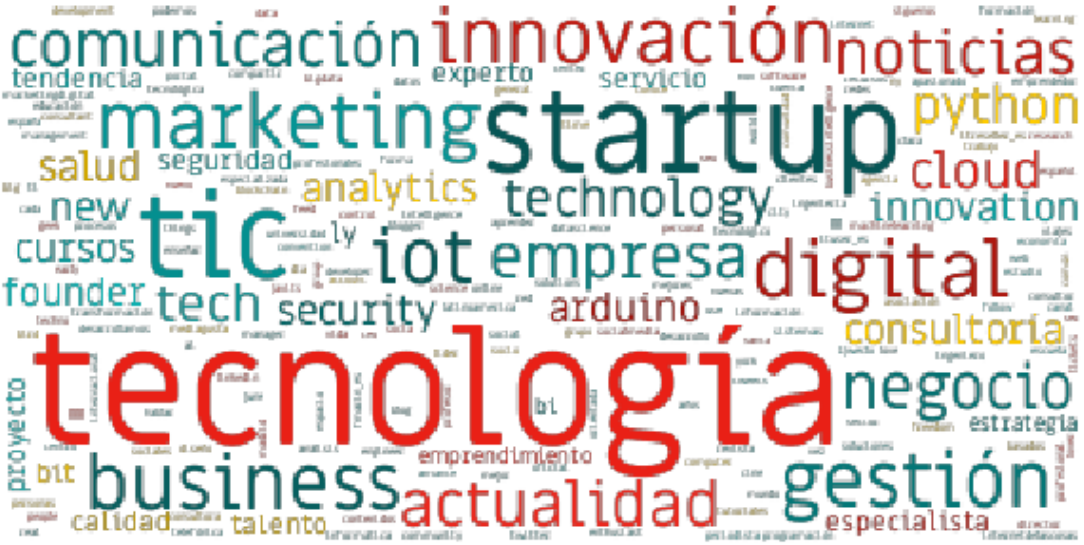


Figura 4.4.12: Wordcloud de las descripciones de los usuarios.

Figura 4.4.13: Wordcloud de las descripciones de los usuarios.

Aparentemente, las descripciones de los usuarios coinciden con los perfiles que nos interesan.

Origen de los tuits

Otro elemento que nos va a proporcionar información relevante para nuestro objetivo es la aplicación de origen del tuit. Un tuit emitido desde un dispositivo móvil es muy probable que haya sido emitido por una persona y no por un bot o una empresa. Entre los campos del tuit, hay uno que refleja esta información: "source"[4]. Este campo describe la utilidad empleada para publicar el tuit y está consignada en formato HTML. Los tuits publicados desde la web de la aplicación tienen un valor en el campo "web".

[4] https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/tweet-object

A continuación describimos los principales orígenes de los tuits almacenados en nuestra base de datos, y al igual que con los hashtags, comparamos los de los tuits descargados, y los de aquellos citados o retuiteados en ellos:

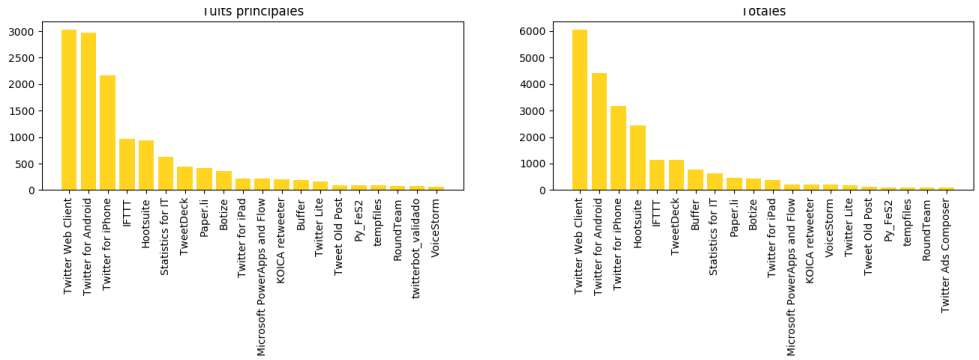


Figura 4.4.14: Orígenes más frecuentes en los tuits descargados (izq.) y en los tuits descargados y citados y retuiteados en ellos.

Estos orígenes quieren decir:

- Twitter web client: son tuits producidos desde la página web de Twitter
- Twitter for Android, Twitter for iPhone, Twitter for iPad: tuits enviados desde las aplicaciones de Twitter para los sistemas operativos de los dispositivos móviles.
- IFTTT, Hootsuite, Paper.li, RoundTeam: son herramientas que sirven para gestionar la actividad en plataformas sociales, por ejemplo publicar automáticamente contenidos (Twitter entre otras).
- Statistics for IT, Koica retweeter: parecen retuits automáticos por parte de alguna organización.
- TweetDeck es una herramienta oficial creada por Twitter para gestionar y controlar varias cuentas desde un solo lugar. Se pueden controlar notificaciones, menciones, mensajes y la actividad en general de una o varias cuentas, y por supuesto programar envíos de tuits.
- Botize es una herramienta para automatizar tareas, en particular en Twitter. Los tuits con este origen serán probablemente de bots.
- PowerApps and Flow: PowerApps es un servicio de Microsoft orientado fundamentalmente a empresas para crear aplicaciones. Microsoft Flow es un servicio también enfocado sobre todo a empresas, para principalmente ayudarles a automatizar tareas entre sus sistemas y algunos de terceros (Twitter entre ellos).
- Twitter Lite: es una versión oficial de Twitter, enfocada sobre todo a usuarios de países emergentes, que funciona a través del explorador y que minimiza el uso de datos y mejora la velocidad de carga en conexiones lentas.

- Tweet Old Post: es un plugin de WordPress, que retuitea de forma aleatoria y automática entradas antiguas de un blog alojado en dicha plataforma.
- Buffer, Py_FeS2, tempfiles, twitterbot_validado: no está muy claro lo que representan. En Twitter, @Py_FeS2 es un usuario nuevo, de Noviembre de 2017, que por el volumen tuiteado parece un bot.
- VoiceStorm: es una aplicación que permite a los empleados de una empresa promocionar la entidad en la que trabajan a través de las redes sociales.

De este análisis parece que si un tuit proviene de un origen como Twitter web client, Twitter for Android, Twitter for iPhone, Twitter for iPad, Twitter Lite, Tweet Old Post y VoiceStorm, es bastante probable que el usuario sea una persona (y por tanto un posible usuario relevante para el objetivo del proyecto). A continuación mostramos los tuits clasificados como "posibles personas" y "posibles otros" atendiendo a este criterio:

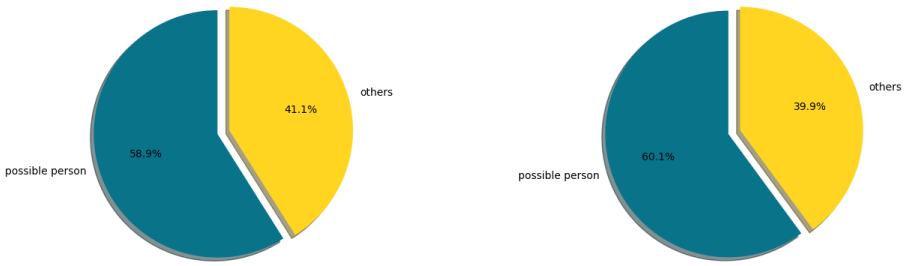


Figura 4.4.15: Porcentaje en los datos de los tuits cuyo origen indican posibles personas en los tuits descargados (izq.) y en los tuits descargados y citados y retuiteados en ellos.

Capítulo 5

Limpieza de datos: extracción de usuarios relevantes

Como hemos visto en el análisis exploratorio inicial de los datos, la fase de limpieza de la información va a ser muy relevante para conseguir nuestro objetivo final. Se trata de extraer, a partir de los tuits almacenados, una lista de usuarios que pudieran constituirse en candidatos adecuados a una oferta de trabajo (como, por ejemplo, las de la figura 1.1.2). Para ello, de todos aquellos usuarios de los que tenemos constancia en los tuits recogidos hemos de seleccionar aquellos que sean personas (eliminando bots, empresas, etc.) y que hayan publicado contenido relacionado con la materia de referencia, en este caso la ciencia de datos.

Los desarrollos descritos en este capítulo están en el fichero **seleccion_usuarios.py**, que puede encontrarse en el repositorio de GitHub.

5.1 Detección del idioma

El primer paso para poder analizar el contenido de un tuit y determinar si dicho tuit (y por tanto el usuario que lo ha publicado) está relacionado con la ciencia de datos o el big data, es determinar el lenguaje en el que está escrito. Como ya hemos mencionado, aunque la búsqueda en Twitter se realizó solicitando el campo "languages = ["es"]", obtenemos algunos tuits en otros idiomas, como estos dos que mostramos a continuación:

Figura 5.1.1: Tuits no solo en español.

Los primeros trabajos sobre el problema de la detección automática del lenguaje de un texto se remontan a la década de 1970 [9]. En la mayoría de las propuestas, existe una fase de entrenamiento, sobre textos previamente clasificados, en la que se produce un modelo del lenguaje (tal vez uno por lenguaje), y una fase de reconocimiento, en la que el lenguaje de mayor verosimilitud para el texto se extrae a partir de la aplicación de los distintos modelos. La clave de todos estos métodos es la modelización del lenguaje, algo que puede conseguirse atendiendo a diversas características diferenciadoras: fonemas, morfología, sintaxis y/o prosodia.

La aplicación de dichas técnicas a textos provenientes de entornos web, blogs, foros, etc. no está exenta de problemas, ver por ejemplo [10]. Los textos procedentes de entornos web en general, y de Twitter en particular, presentan elevados niveles de lo que podríamos denominar como "ruido", por ejemplo:

- suelen ser textos cortos, lo que dificulta la aplicación de técnicas basadas en frecuencia de palabras o caracteres,
- presencia de enlaces web, etiquetas, emoticonos y otros caracteres propios del entorno,
- uso de jerga, lenguaje informal y palabras en idiomas distintos del principal del texto,
- modificaciones de la ortografía, que van desde palabras abreviadas ("q", "xa") a expresiones enfáticas ("mooooooolaaaaaaaa"), por poner dos ejemplos.

A pesar de que es posible encontrar corpus de textos con estas características ya clasificados por idioma[1] que pudieran servir para entrenar un modelo que determinara el lenguaje, hemos optado por usar un clasificador que no necesite entrenar un modelo, ya que esta parte del proyecto no es la principal. Hemos encontrado referencias, como [10], que apuntan al buen comportamiento de clasificadores que no depende de conjuntos de entrenamiento (basados en "small words", también denominadas "stop words", y trigramas). Pero este método requiere de una implementación *ad-hoc*, y las opciones disponibles, listas para usar, nos han parecido lo suficientemente buenas para el objetivo que perseguimos en este apartado.

[1] https://blog.twitter.com/engineering/en_us/a/2015/evaluating-language-identification-performance.html

En el entorno de Python, hemos encontrado varios paquetes que tratan el problema de detectar el idioma de un texto automáticamente. Los más comentados son los siguientes:

- `langid` [11]: es un paquete que proporciona un clasificador Naïve Bayes de textos, que usa n-gramas (secuencias de n caracteres en el texto, con $1 \leq n \leq 4$). El clasificador está pre-entrenado sobre diversos corpus de texto, en un total de 97 idiomas. Según los resultados explicados en el artículo de presentación del trabajo, es un método con una exactitud (*accuracy*) del 94%.

- `langdetect`: de Nakatani Shuyo, también es un clasificador Naïve Bayes basado en n-gramas, con normalizaciones heurísticas, https://github.com/shuyo/language-detection.

- `LDIG`: es un clasificador específico para Twitter, creado por el autor de *lagdetect* para solventar las carencias de éste en la clasificación de mensajes cortos, https://github.com/shuyo/ldig. Soporta menos idiomas que los anteriores.

- `equilid`: es el paquete de más reciente creación que hemos encontrado, que además está especialmente diseñado para tratar el "ruido" que comentábamos caracteriza los textos de Twitter. Está concebido para identificar dialectos urbanos y tratar correctamente expresiones de jerga.

Dado el tema que nos ocupa, relacionado con ciencia de datos, la especialización de `equilid` y la dificultad de su instalación en el sistema disponible (usa una versión de TensorFlow que aparentemente no está desarrollada para Windows), nos hace descartarlo de entrada. En [12] se muestra que el comportamiento de cualquiera de los clasificadores restantes es lo suficientemente bueno por separado para el objetivo de este proyecto. Sugieren que combinar dos o más clasificadores puede mejorar la exactitud de la clasificación, y que en general, la limpieza de los tuits (urls, etiquetas, menciones, etc.), si bien mínimamente en algunos casos, suele mejorar el comportamiento de los modelos (excepto en el caso de `langid`). Finalmente, nos hemos decidido por usar el algoritmo del paquete `langid`, visto el buen resultado reportado, y la facilidad de uso del mismo.

En [12] se muestra que, en un contexto general de detección del lenguaje, el paquete `langid` no parece beneficiarse de una limpieza del tuit para retirar urls, menciones, etiquetas y emoticonos del cuerpo del mensaje. Sin embargo, dado que la implementación de esa limpieza no es difícil, hemos comparado la clasificación del lenguaje obtenido con el texto original y con el texto limpio, y hemos observado que en los textos de los $24,128$ tuits descargados (contando retuiteados y citados también) hay un 6.37% de tuits que no tienen la misma asignación de lenguaje. Estos tuits suelen ser tuits con pocas palabras o con mucha mezcla de idiomas (generalmente español e inglés).

Como método alternativo, se ha implementado un clasificador manual que solo usa "stop words" para tratar esos casos en los que `langid` no da una elección clara del idioma. Si el método de los "stop words" da un idioma para el texto que coincide con alguno de los proporcionados por `langid` (bien el del texto limpio, bien el del texto original), ese será el que se asigne al tuit. Y si no, lo dejaremos clasificado con un idioma desconocido.

Hemos aplicado este método de clasificación del lenguaje a los $24,128$ textos de los tuits originales, retuiteados o citados, y resulta una clasificación en 25 idiomas diferentes. Mostramos a continuación un resumen de los resultados del clasificador:

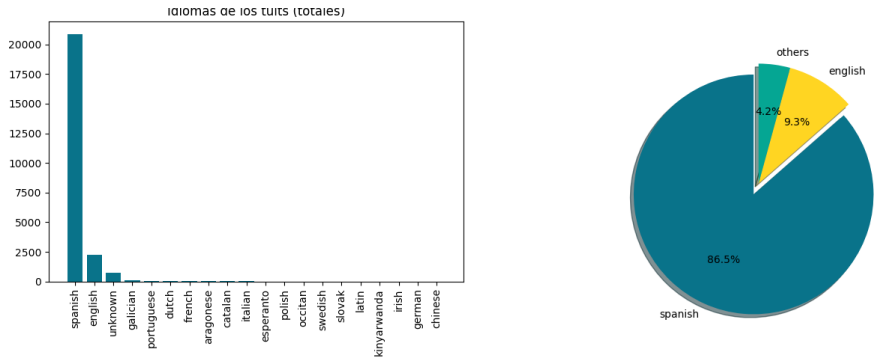


Figura 5.1.2: Clasificación por lenguaje del texto del tuit. Del 4.2% de "others" hay un 2.9% de "unknown". ¿Errores de clasificación?

Echando un vistazo a los resultados, aquellos que no están clasificados o que están clasificados en idiomas poco probables, digamos, como el esperanto, son aquellos en los que el clasificador seguramente ha fallado. Algunas veces parece ser porque casi no tuvieran texto o porque el texto contenga mucha mezcla de idiomas. Pero otras veces, como en el siguiente caso clasificado en esperanto, no está muy claro por qué:

Figura 5.1.3: Tuit clasificado como esperanto.

5.2 Tipo de usuario

Para clasificar al usuario respecto a su entidad, y por ejemplo distinguir entre personas, bots y empresas, nos parece que la parte del tuit que más información contiene, a partir de los datos conseguidos, es la bio que los propios usuarios aportan. Antes de cualquier labor de análisis de esos textos, es necesario saber el idioma en el que están, y por ello de nuevo aplicamos a nuestros datos el identificador de lenguaje que usamos en el apartado anterior.

Primero seleccionamos los datos correspondientes a los usuarios distintos, obteniendo $7,210$ usuarios con distinto "*id_str*". `langid` clasifica de forma diferente el idioma de los datos de perfil antes y después de limpiarlos (quitar urls, emojis, hashtags, etc.) en un 5.99% de los casos. Después de aplicar en éstos últimos el método de las "stop words", los perfiles de los usuarios han quedado clasificados en 44 idiomas diferentes.

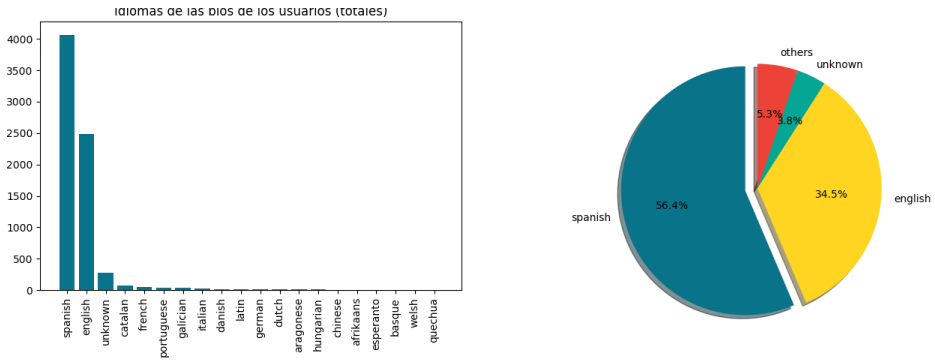


Figura 5.2.1: Clasificación por lenguaje del texto de las bios.

5.3 Naturaleza del tuit

texto del tuit: IT, cientifico, analista o nodatascience (diccionarios de palabras) /Binario, data science o no data science

Capítulo 6

Modelado de los datos: ordenación de los usuarios

6.1 Principales hipótesis

6.2 Algoritmos

6.3 Almacenamiento

Capítulo 7

Visualización de los resultados

7.1 Herramientas

7.2 Acceso web

Capítulo 8

Áreas de mejora

Bibliografía

[1] María Gloria Castaño collado, Gerardo de la Merced López Montalvo, José María Prieto Zamora, *Guía técnica y de buenas prácticas en reclutamiento y selección de personal (R& S)*. Documento aprobado por la Junta de Gobierno del Colegio Oficial de Psicólogos de Madrid, Febrero de 2011. http://www.copmadrid.org/webcopm/recursos/guiatecnicabuenaspracticas.pdf

[2] *Selección de personal para no especialistas*. Andalucía Emprende, Fundación Pública Andaluza. Consejería de Economía y Conocimiento. https://www.andaluciaemprende.es/wp-content/uploads/2015/02/guia\discretionary{-}{}{}seleccion-personal.pdf

[3] *Ley Orgánica 15/1999, de 13 de diciembre, de Protección de Datos de Carácter Personal*. Jefatura del Estado BOE núm. 298, de 14 de diciembre de 1999 Referencia: BOE-A-1999-23750 http://www.agpd.es/portalwebAGPD/canaldocumentacion/legislacion/estatal/common/pdfs/2014/Ley_Organica_15-1999_de_13_de_diciembre_de_Proteccion_de_Datos_Consolidado.pdf

[4] María Luz Congosto Martínez, *Caracterización de usuarios y propagación de mensajes en Twitter en el entorno de temas sociales*. Tesis doctoral.

[5] "Twitter". Wikipedia. https://es.wikipedia.org/wiki/Twitter.

[6] Shamanth Kumar, Fred Morstatter, Huan Liu. *Twitter Data Analytics*. Springer (2013).

[7] Twitter Developer Dpcumentationhttps://dev.twitter.com/

[8] Steven Bird, Ewan Klein, Edward Loper. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly (2009). http://www.nltk.org/book/

[9] Marc A. Zissman, Kay M.Berkling. Automatic language identification. *Speech Communication*, Volume 35, Issues 1–2, August 2001, Pg. 115-124.

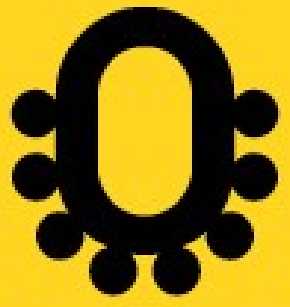
[10] Y. Almeida-Cruz, S. Estévez-Velarde, A. Piad-Morffis. Detección de Idioma en Twitter. *GECONTEC: Revista Internacional de Gestión del Conocimiento y la Tecnología*, Vol.2 (3), 2014. https://www.upo.es/revistas/index.php/gecontec/article/view/1081/pdf_11

[11] Marco Lui, Timothy Baldwin. `langid.py`: An Off-the-shelf Language Identification Tool. *Proceedings of the ACL 2012 System Demonstrations*, pg. 25–30, 2012. http://www.aclweb.org/anthology/P12-3005

[12] Marco Lui, Timothy Baldwin. Accurate Language Identification of Twitter Messages. *Proceedings of the 5th Workshop on Language Analysis for Social Media (LASM) @ EACL 2014*, pg. 17-–25, (2014). http://www.aclweb.org/anthology/W14-1303

[13] David Jurgens, Yulia Tsvetkov, Dan Jurafsky. Incorporating Dialectal Variability for Socially Equitable Language Identification. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Short Papers)*, pg. 51-–57, (2017). https://doi.org/10.18653/v1/P17-2009

Documento producido con LaTeX.



OCTOPUS

DATA INSIGHTS