Escuela Politécnica Superior

Análisis de la credibilidad de cuentas en Twitter

Grado en Ingeniería Informática

Trabajo Fin de Grado

Autor:
María Esther Rico Martínez

Tutor/es:
José Vicente Berná Martínez

Diciembre 2019

Universitat d'Alacant
Universidad de Alicante

Resumen

Hoy en día las redes sociales están muy presentes en nuestro día a día, habiéndose convertido en unas plataformas donde las personas pueden informar o informarse sobre gran variedad de noticias, acontecimientos y/o compartir sus propias ideas. Sin embargo, la facilidad con la que se hace viral la información en la red llama, en numerosas ocasiones, a la aparición de cuentas falsas con intención de desinformar o de engañar. Por ello, no es recomendable darle total credibilidad a cualquier publicación que leamos, pues es necesario valorar quién es el usuario autor de dicho contenido y si se trata de una fuente fiable o, por el contrario, inspira poca confianza.

Así pues, este trabajo pretende colaborar con esta ardua tarea de investigar la veracidad de un contenido y de su autor, aportando para ello un algoritmo capaz de puntuar a los usuarios de Twitter en base a la credibilidad que estos presenten según el análisis de determinadas características presentes en sus perfiles. De esta forma, según los valores que muestren estos indicadores seleccionados, se les otorgará de manera individual mayor o menor puntuación de fiabilidad a cada uno de ellos.

Una vez calculadas estas puntuaciones, para cada uno de los aspectos de la cuenta que se consideran relevantes, se procede a calcular la valoración final de credibilidad que presenta el usuario. De esta manera, cuanta mayor sea la puntuación final obtenida por un perfil de usuario, mayor fiabilidad presentará, disminuyendo así la posibilidad de que se trate de una cuenta controlada por algún tipo de software (bot) o de una cuenta falsa.

Estos indicadores a estudiar se obtienen usando la API de Twitter que, junto con algunas librerías del lenguaje Python, permiten extraer los datos de cada perfil que se consideren necesarios para realizar esta valoración de credibilidad. Sin embargo, no todos los indicadores que se extraen de un perfil repercuten de la misma forma en esta puntuación final, pues habrá algunos más determinantes que otros. Así pues, aquellos indicadores que resulten ser más determinantes a la hora de valorar la credibilidad de un usuario presentarán puntuaciones base más influyentes que las no determinantes.

Sin embargo, estudiar la credibilidad de un usuario en un momento determinado, será mucho menos fiable que si se hace un estudio de dicha fiabilidad a lo largo del tiempo. Así pues, cuanto más amplio sea el periodo de tiempo dedicado al análisis de un perfil, más fiable será el resultado obtenido y permitirá estudiar la tendencia de credibilidad con mayor exactitud.

Motivación, justificación y objetivo general

El origen de este proyecto parte del gran interés que me despierta la aplicación del Big Data en el mundo real, ya que cursando el itinerario de Sistemas de la Información he podido descubrir lo realmente importante y necesario que resulta el poder extraer conclusiones relevantes analizando grandes volúmenes de datos.

Así pues, para poder catalogar la veracidad de la información que poseemos y que va a ser sujeto de estudio, es necesario investigar con detenimiento el origen de esta, ya que cada vez es más frecuente encontrar datos falsos difundidos en la red con el fin de engañar o desinformar. Esto supone una importante amenaza sobre la neutralidad de la red, viéndose agravado por la gran facilidad de difusión que presenta cualquier tipo de información.

Debido a la fuerte presión social que ejercen estas redes sobre la población, es de especial importancia tener cautela sobre la información que se propaga, ya que puede desencadenar errónea e intencionadamente movimientos extremistas y/o radicales sobre la sociedad. Además, el hecho de que haya tanta información de dudosa procedencia llega a causar incertidumbre y desconfianza sobre la fiabilidad de las noticias que se publican, llegando a ser muy difícil distinguir los hechos falsos de los reales.

Por ello, quiero centrar mi estudio en el análisis de la veracidad de la información publicada en una red social concreta, diseñando para ello un algoritmo que pondere la misma en base a su autor, los seguidores de este, su impacto en la red social y su actividad.

Para afianzar la viabilidad de este proyecto, pretendo hacer uso de distintas APIs que faciliten la recogida de datos de la red social Twitter sobre la cual se busca enfocar este estudio. De esta manera, pretendo obtener y ponderar varias muestras de publicaciones presentes en esta red social con el fin de determinar y puntuar la credibilidad de los hechos y extraer conclusiones con las que poder detectar noticias ficticias o falseadas, también conocidas con el nombre de “fake news”.

Considero que desarrollar esta solución al problema planteado contribuirá a demostrar mis habilidades como futura ingeniera, ya que la sociedad debe tener acceso a una información veraz, de calidad y real. Una noticia ha de contar los distintos hechos o sucesos que ocurren en la actualidad, sin temer que aquello que estamos leyendo resulte ser fruto de la falsedad y/o de la intención de influir interesadamente en la conducta de la sociedad.

Agradecimientos

Me gustaría agradecerle especialmente a mi tutor de TFG, José Vicente Berná, toda su ayuda, consejos, sugerencias e ideas que me ha brindado a lo largo de las distintas fases de mi proyecto. Quiero darle las gracias además por su enorme comprensión y apoyo cuando por circunstancias ajenas y personales no he podido entregarle avances de mi trabajo dentro de los plazos previstos.

Gracias de verdad. Valoro muchísimo la paciencia que has tenido y, en especial, estas últimas semanas que has estado ayudándome mucho a orientar y pulir mi trabajo.

Además, me gustaría agradecer a mi familia y a mi pareja el apoyo que me han brindado constantemente, tanto en los buenos momentos como en los malos, animándome en todos esos momentos de flaqueza y mostrándome su apoyo incondicional ante todas las metas y objetivos que me propusiera.

Citas

La gente piensa que enfocarse significa decir sí a aquello en lo que te enfocas, pero no es así. Significa decir no a otros cientos de ideas buenas que hay.

Steve Jobs

La imaginación es la facultad del descubrimiento, preeminentemente. Es lo que penetra en los mundos nunca vistos a nuestro alrededor, los mundos de la ciencia.

Ada Lovelace

Ser inundado con información no quiere decir que tengamos la información correcta o que estemos en contacto con las personas correctas.

Bill Gates

Uno no debería tener miedo a decir "no lo sé" o "no lo entiendo", o incluso de hacer "preguntas tontas". Ninguna pregunta es tonta. Aunque las cosas puedan parecer imposibles, aunque los expertos digan que algo es imposible, aunque haya que seguir el camino sola, no hay que tener miedo a estar equivocada, a admitir errores; aquellos que sepan fallar de forma estrepitosa son los que pueden conseguir cosas grandiosas.

Margaret Hamilton

Índice de contenidos

Índice de figuras

Índice de tablas

1. Introducción

Las redes sociales originalmente surgen como un medio de comunicación que, mediante el uso de las nuevas tecnologías, nos ofrecen la posibilidad de comunicarnos, expresar nuestras ideas o compartir información de forma casi instantánea con uno o varios usuarios vía Internet, ya bien sea de forma directa como es el caso de los sistemas de mensajería (WhatsApp, Telegram, Line…) o bien de forma indirecta mediante publicaciones (Twitter, Facebook, MySpace...).

Hoy en día las redes sociales forman parte de nuestra rutina, ya que hacemos uso de ellas en diversas situaciones a lo largo del transcurso del día y, de manera equivocada, confiamos muchas veces en la veracidad de las publicaciones que leemos sin corroborarla. La adaptación de las nuevas tecnologías sobre la sociedad y su bajada de precios han originado que disponer de un terminal con conexión a internet no suponga una dificultad en la actualidad, agravando consigo esta dependencia con el paso de los años.

De esta forma, la fuerte conectividad social y la rápida propagación de la información han llegado a desfigurar el objetivo inicial de las redes sociales de ser un medio de comunicación, surgiendo así nuevos propósitos (sociales, políticos, empresariales…). Este hecho ha fomentado, entre otras cosas, la divulgación de manera interesada de información falseada en la red de manera viral.

Un claro ejemplo sería el caso de la red social Twitter, pues cualquier persona puede hoy por hoy crearse una cuenta de usuario y publicar “tweets” con información manipulada buscando generar determinados movimientos sociales sobre la comunidad, ya que mediante la opción “retwittear” una publicación puede propagarse a gran velocidad dentro de esta red social.

Con el paso del tiempo han ido apareciendo cada vez más publicaciones falsas que se propagan a gran velocidad en internet [22], creadas con la única intención de propagar hechos irreales pero que para determinados organismos sociales, políticos o empresariales puede suponer algún tipo de ventaja competitiva.

Con ello, surge el término de “fake news”, las cuales resultan ser noticias falseadas o alteradas con el único fin de desinformar o engañar a la población con finalidades específicas. Este hecho favorece enormemente a determinadas causas pertenecientes, en especial, a los ámbitos empresarial y/o político. Por ello, resulta realmente necesario encontrar la manera de poder filtrar todo este tipo de informaciones falsas y tener como ciudadanos, derecho a disponer de información veraz y real.

Así pues, en este proyecto se busca desarrollar una solución con la que poder reconocer noticias falsas con el fin de filtrar las mismas y así descatalogar este tipo de información falseada, colaborando de esta manera a limpiar paulatinamente la red y a favorecer la veracidad en la misma.

Las empresas u organismos políticos que de verdad apoyen la propagación de noticias reales, podrán tener acceso a una solución que colabore con la divulgación de información objetiva. De esta manera, se devolverá a la población la posibilidad de definir sus ideas y/o gustos personales sin verse influenciados por noticias falsas o tendenciosas.

2. Planificación

La planificación contemplada que se indica más abajo la Tabla 1 sobre cómo se pretende organizar el desarrollo de este proyecto consta de 3 grandes bloques:

- **En primer lugar**, se comenzará plasmando la motivación que ha llevado a realizar este trabajo, sobre qué tema se busca profundizar y las posibles repercusiones y/o beneficios que ello puede traer.

  Seguidamente, se comenzará con la introducción donde se ubicará el contexto sobre el cual va a girar el proyecto y su situación en la actualidad.

  Una vez contextualizado y definido el marco sobre el cual se pretende investigar, se procederá a realizar el apartado del estado del arte, donde se hará estudio tanto de los conceptos que intervienen en el marco del problema, como de tecnologías, trabajos y/o estudios ya existentes sobre el tema que puedan servir como referencia para comenzar.
- **En segundo lugar**, una vez se ha estudiado sobre el problema, su contexto y posibles soluciones, se procede a establecer los objetivos que se pretenden conseguir con el desarrollo del proyecto. Una vez establecidos, se propone una planificación a seguir para cumplir cada uno de estos puntos.

  Finalmente se propone una posible solución que aborde el problema y posteriormente se validará. Para esta validación se hará una recogida de datos durante 1 mes.
- **En tercer lugar**, se procederá a implementar la solución y a validar el correcto funcionamiento del código implementado.

  Tras ello, una vez comprobado que lo implementado funciona correctamente será momento de realizar pruebas para validar y afianzar que aquello que se ha desarrollado funciona en mayor o menor medida. Para esta validación se hará una recogida de datos durante 1 semana.

  Seguidamente se detallarán las conclusiones que se extraen del estudio y posibles mejoras, variantes o trabajos futuros que podrían llevarse a cabo realizado con ello.

  Finalmente, se procederá a revisar que todas las referencias añadidas en el documento están correctamente enlazadas, así como completar el apartado de citas y agradecimientos.

Tabla 1. Planificación temporal TFG

Contenidos	**Tiempo específico**	**Tiempo total**	**Fecha límite fin**
Motivación, justificación y objetivo general Introducción Estado del arte		1 mes	30 septiembre
Objetivos Planificación		1 mes y medio	31 octubre
Propuesta y validación			
• Recogida de datos para validar la propuesta	1 mes		
Implementación y validación		1 mes	20 noviembre
• Recogida de datos para hacer las pruebas	1 semana		
Pruebas Conclusiones y trabajo futuro Referencias, bibliografía y apéndices Agradecimientos, citas, índices			

3. Estado del arte.

3.1. Noticias falsas en las redes sociales

Hoy en día cada vez es más frecuente desconfiar de la información ofrecida por los medios de comunicación o las redes sociales. Con la llegada de estas últimas, la posibilidad de poder viralizar cualquier tipo de información supone un gran problema a la hora de distinguir lo real de lo ficticio. En la actualidad, desinformar se ha convertido en una de las principales herramientas de muchos organismos en busca de beneficios comerciales, políticos y/o sociales. Este tipo de información falseada o también conocida con el nombre de "fake news", ejerce una influencia sobre la sociedad que resulta alarmante y, por ello, redes sociales como Facebook, WhatsApp o Twitter trabajan en soluciones que puedan frenar está difusión de contenido fraudulento, empleando distintos métodos para ello. WhatsApp recientemente ha comenzado a restringir el reenvío de un mismo mensaje hasta 5 veces como máximo [1]. Facebook en cambio opta por apostar por servicios verificadores de datos [2], mientras que Twitter estudia condicionar la difusión de tweets de una persona en base al comportamiento de los usuarios que forman parte de su red [3].

Al margen de las medidas que están adaptando las distintas redes sociales, los usuarios de estas juegan un papel también importante a la hora de salvaguardar la integridad y veracidad de los datos publicados en estas [4]. En muchas ocasiones en las cuales se ha publicado contenido falso, muchos usuarios lo han compartido o difundido asumiendo que aquello que estaban leyendo era real. Por ello, es necesario analizar cuidadosamente la fiabilidad del autor de la publicación y realizar un contraste de dicha noticia en diferentes fuentes reconocidas. En el caso de determinar que una noticia no es fiel a la realidad, los usuarios deberían colaborar en su reporte, a fin de que la plataforma tome medidas para la eliminación de dicho contenido y así detener su propagación.

Sin embargo, para poder ser capaces de identificar noticias falsas, los usuarios deben de contar con un mínimo de conocimiento sobre la publicación a cuestionar y tener capacidad crítica con la que descartar este tipo de información falseada. Además, estudios recientes han confirmado que dependiendo de la zona geográfica se pueden encontrar diferencias significativas a la hora de identificar contenido fraudulento, pues hechos como la desinformación mediática en un país o la escasa alfabetización dificultan esta tarea [5].

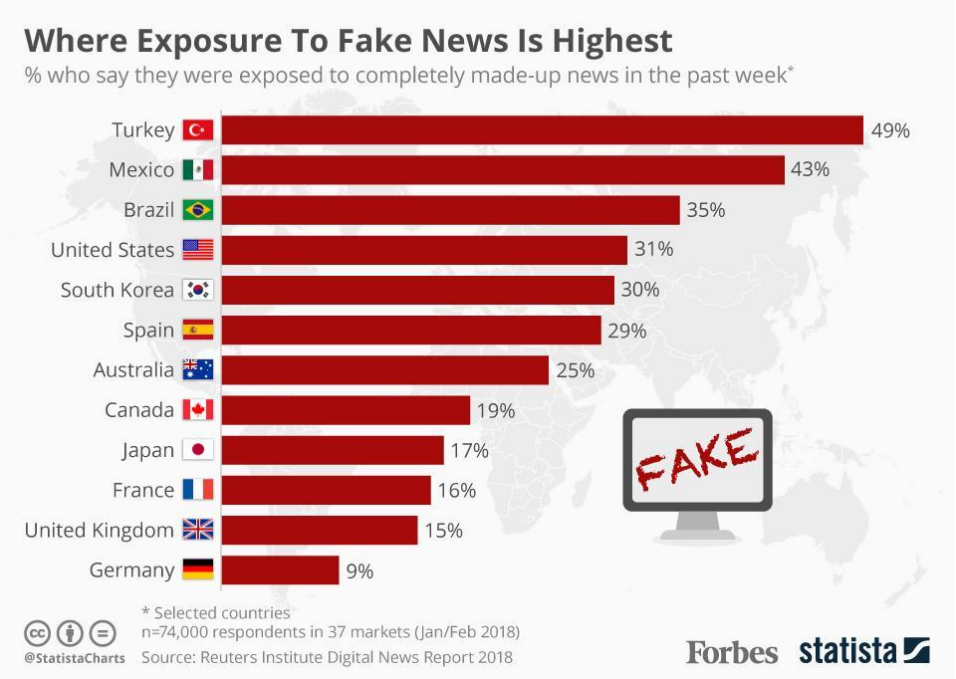


Figura 1. Gráfico sobre la exposición a las "fake news" según el país (2018) (Fuente https://www.forbes.com/)

Todo ello ha propiciado la aparición de perfiles falsos, algunos de los cuales esconden detrás un software programado para simular comportamiento y/o acciones humanas, también conocido con el nombre de "bot". Aunque se trata de un hecho relevante, en la actualidad no supone más de un 5% la cantidad de cuentas falsas que existen en la red social Twitter, a diferencia de la red social Facebook que presenta en la actualidad aproximadamente 83 millones de cuentas falsas [21].

El hecho de poder difundir tan rápidamente información ha desencadenado cada vez más la aparición de contenido falso como resultado de modificar datos reales en beneficio propio [31] o por el simple hecho de desinformar [32].

Además, un reciente estudio realizado por el Instituto de Tecnología de Massachusetts (MIT) sobre la red social Twitter, confirma que las noticias falsas resultan más atractivas y/o novedosas para los lectores y, en consecuencia, se propagan con mayor rapidez que cuando se trata de contenido real. De esta forma, el contenido falso resultaba ser un 70% más propenso a difundirse mientras que las publicaciones reales rara vez eran compartidas por más de 1000 cuentas. Para una mayor certeza, los investigadores filtraron aquellas cuentas que resultaban ser "bots". Sin embargo, pronto se dieron cuenta que tanto con ellos como sin ellos el estudio desvelaba la misma conclusión: las cuentas causantes de la mayor parte de esta difusión de contenido falso eran de personas reales [22].

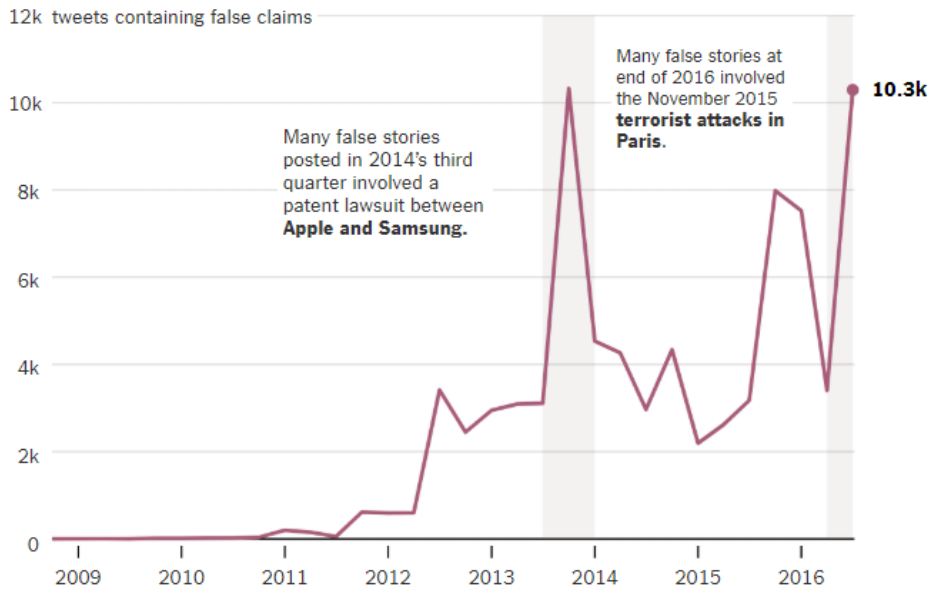


Figura 2. Gráfico sobre el aumento de publicaciones falsas en Twitter. (Fuente https://www.nytimes.com/)

3.2. Twitter

Twitter es una red social nacida en 2006 a manos de Jack Dorsey, Biz Stone, Noah Glass y Evan Williams. Su objetivo principal resulta ser la interacción de sus usuarios mediante la creación y divulgación de contenidos en publicaciones denominadas "tweets".

Inicialmente estas publicaciones, poseían una restricción de 140 caracteres que, posteriormente en el año 2017, se aumentó a 280 derivado de la petición popular. El hecho de que se pudiera compartir información en tiempo real supuso un gran avance en la propagación de noticias de última hora, ya que cualquier suceso que tuviera lugar en un determinado instante podía anunciarse y propagarse de manera instantánea en la red social. Dicha instantaneidad a la hora de publicar cualquier suceso o noticia hace que resulte uno de los medios más atractivos para la prensa.

En la actualidad, Twitter presenta más de 300 millones de usuarios, pudiéndola encontrar entre las 15 redes sociales con más usuarios activos del mundo.

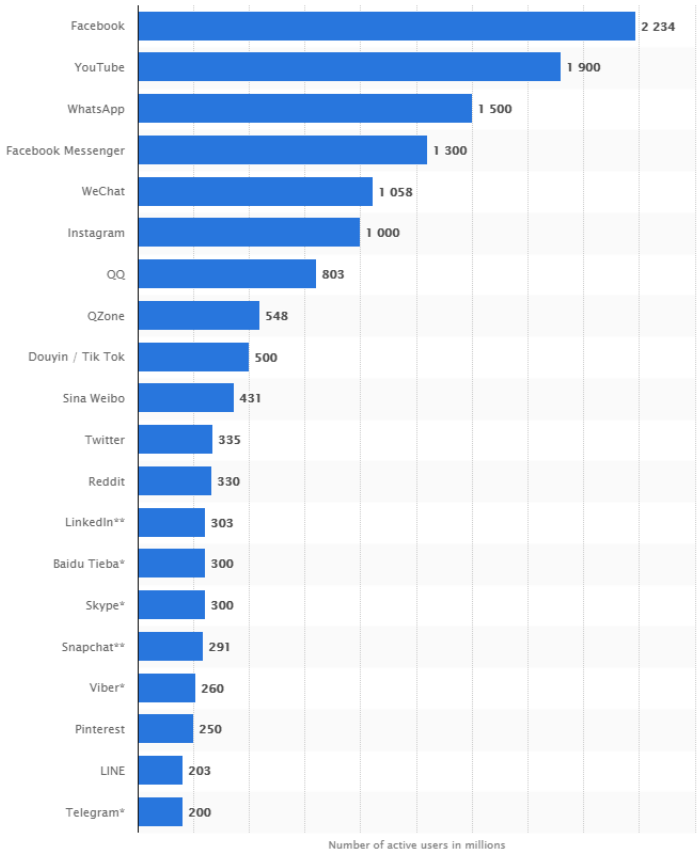


Figura 3. Gráfico sobre el número de usuarios activos en las redes sociales a nivel mundial (2018) (Fuente https://www.statista.com/)

Además, tal y como se indica en la guía de Twitter [6], la comunicación en esta red social comienza con una simple publicación o "tweet". Como usuarios, si una determinada publicación nos agrada podemos indicarlo usando la opción "marcar como favorito" o añadir dicha publicación para que aparezca en nuestro perfil mediante la opción "retweet". Por otra parte, existe la opción "seguir" destinada a agregar al autor cuyo contenido nos sea de interés con el fin de estar informados de todas y cada una de las publicaciones que realice este usuario. Además, la red social dispone también de un buscador por palabras con el fin de filtrar contenido concreto sobre aquello que estemos buscando. Otra opción que nos brinda Twitter, que es la más reciente, se denomina "hashtag" y consiste en añadir etiquetas precedidas por '#' a las publicaciones. De tal manera que, además de poder utilizar el buscador por palabras, podremos también buscar y filtrar el contenido mediante estos "hashtags" o etiquetas.

Con el transcurso del tiempo, distintas empresas y organizaciones [34], han visto en Twitter, gracias a los servicios para negocios que ofrece la plataforma, una manera de lanzar campañas, expandir su influencia y/o una solución para captar las opiniones, preferencias o gustos de las personas haciendo uso de las estadísticas de la red social mediante el servicio Twitter Analytics

[23]. Estos datos estadísticos les permiten analizar a los usuarios y con ello estudiar la orientación de distintas campañas o anuncios con el fin de potenciar y aumentar la conversión de clientes.

3.3. Detección de contenido falso en las redes sociales

Hoy en día no toda la información publicada en las redes sociales resulta ser fiable y de calidad, ya que en muchas ocasiones puede ser incorrecta o incluso inventada.

Cuando se trata de información compartida en tiempo real, como es el caso de la red social Twitter, el impacto que puede llegar a generar la difusión de contenido malicioso es realmente alarmante, ya que difundir rumores o hechos inciertos puede llegar a desencadenar el pánico u otras reacciones adversas en los usuarios. Por tanto, es necesario detectar a tiempo este tipo de información fraudulenta y, por ello, surge la necesidad de buscar y/o crear soluciones que frenen esta difusión descontrolada de información falsa.

De esta manera, podemos encontrar distintas propuestas de soluciones basadas en diversos ámbitos:

- En primer lugar, se destaca una propuesta basada en la detección de imágenes falsas que han sido difundidas en la red social Twitter [24].
  Este análisis inicialmente constó de un contenedor de muestras elaborado con imágenes falsas y reales. Posteriormente, se analizó a un conjunto de todas esas muestras recolectadas aplicando para ello varias técnicas de análisis con el fin de estudiar cada imagen en profundidad, siendo algunas de las técnicas utilizadas las siguientes:
    - **Análisis del usuario que había compartido y/o difundido dicho contenido**, considerando para ello datos como el número de seguidores, el número de personas a las que sigue o la antigüedad que presenta dicha cuenta entre otros.
    - **Análisis en profundidad de la publicación,** considerando para ello datos como el número de favoritos y/o “retweets” recibidos, el número de veces que había sido mencionada dicha publicación, si existían palabras que expresasen sentimientos o la detección de determinados caracteres como interrogantes (para la detección de preguntas) o exclamaciones (para indicar sorpresa o asombro) entre otros.

  Según este estudio, se consideró que la técnica más eficiente fue la segunda, ya que analizando en profundidad un determinado contenido se consiguió una mayor efectividad que simplemente analizando al usuario autor. Sin embargo, hechos como

que una determinada publicación sobre un acontecimiento reciente se posicione en tendencias de Twitter aumentaban la probabilidad de que dicho contenido fuera difundido por el usuario lector independientemente de que siguiera a la cuenta del autor de dicho tweet o no.

Además, otro factor mencionado que ayudaba considerablemente en esta detección de imágenes falsas resultó ser el uso y aplicación de distintos algoritmos de clasificación como por ejemplo el clasificador bayesiano ingenuo o los árboles de clasificación. Una vez son aplicados estos algoritmos, se llevó a cabo un estudio sobre los resultados obtenidos tras aplicar las distintas técnicas anteriormente mencionadas, con el fin de determinar un patrón de comportamiento asociado a la propagación de este tipo de tweets con imágenes falsas.

- En segundo lugar, cabe destacar otra propuesta que consistía en encontrar noticias falsas de manera automática en redes sociales mediante la aplicación de métodos de evaluación basados en [25]:
    - **Analizar la lingüística del contenido** con el fin de reconocer y determinar comportamientos o palabras que indiquen engaño o intención de desinformar, usando para ello aprendizaje automático (machine learning).

      En este caso, la propuesta sugería el análisis de la sintaxis del contenido empleando PCFG (Probability Context Free Grammars). De esta manera, las sentencias se reescriben de manera estructurada formando un árbol de análisis y se le asigna una determinada probabilidad.

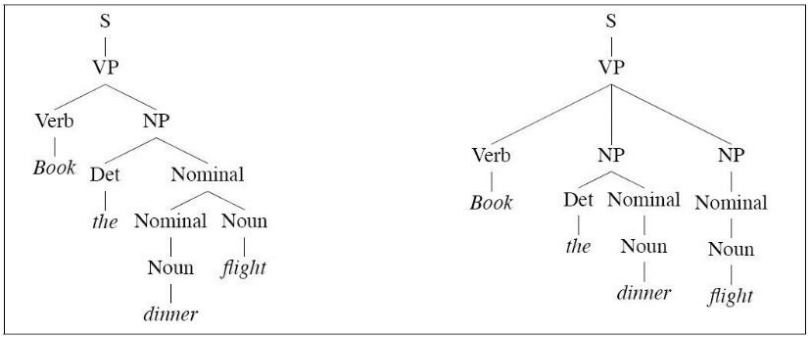


Figura 4. Ejemplos de árboles de análisis empleando PCFG.
(Fuente https://gawron.sdsu.edu/)

    - **Analizar el comportamiento basado en la información de la red**, utilizando para ello los metadatos de la propia publicación y/o estudiando conjuntos de datos

estructurados, ya que facilita la labor de detectar cualquier sintaxis que pueda esconder una intención de engañar o desinformar.

Para ello en el estudio se propuso hacer uso del modo de análisis CRA (Centering Resonance Analysis). Este tipo de análisis representa conjuntos de contenidos de texto identificando aquellas palabras más relevantes que presentan un claro enlace con otros términos o vocablos en la red, generando con ello una red de sustantivos y adjetivos sobre un tema en concreto.

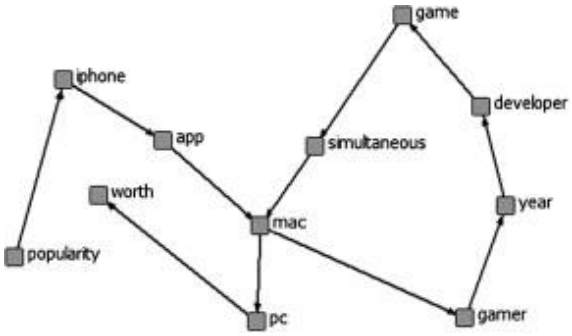


Figura 5. Ejemplo de una red de palabras de un comentario en The Unofficial Apple Weblog (TUAW)
(Fuente https://www.sciencedirect.com/)

Estas soluciones propuestas en el estudio pretendían imitar el juicio empleado en la actualidad por determinados profesionales para la discriminación de noticias falseadas, como editores, reporteros o periodistas entre otros. Así pues, ambos métodos mencionados incorporaban técnicas de aprendizaje automático con el fin de alimentar a los clasificadores para que fueran aprendiendo paulatinamente y adaptándose al análisis en cuestión.

- En tercer lugar, se destaca otra propuesta la cual se basaba en la construcción de modelos de redes neuronales con infusión lingüística con el fin de clasificar un conjunto de publicaciones como noticias reales o sospechosas, propulsando su aprendizaje mediante [26]:
    - **Análisis y aprendizaje del contenido de las publicaciones en Twitter.**
    - **Análisis y aprendizaje de las interacciones sociales** con el fin de clasificar las noticas como reales o sospechosas. Dentro de las noticias sospechosas, el estudio distinguía 4 subtipos principales:
        - Sátira
        - Engaño
        - Clickbait

- Propaganda.

Tras estudiar distintos tipos de arquitecturas de redes neuronales, se observó que aquellas redes que habían sido entrenadas con contenido de Twitter y distintas interacciones sociales presentaban un mejor rendimiento y obtenían mejores resultados que otros modelos.

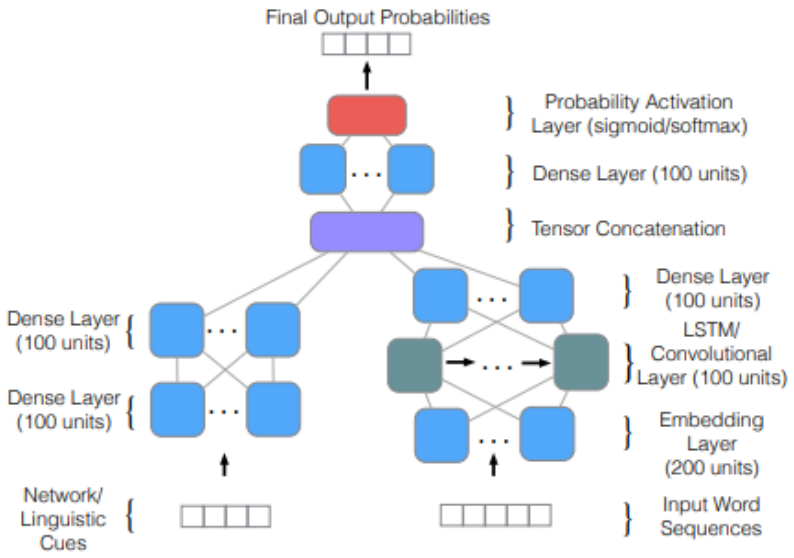


Figura 6. Arquitectura de una red neuronal para la clasificación de noticias basada en las interacciones sociales y en el uso del lenguaje.
(Fuente http://www.aclweb.org/)

Sin embargo, incorporar características lingüísticas como apoyo mejoraba la clasificación de las publicaciones, mientras que las interacciones sociales únicamente contribuían en la diferenciación de los 4 subtipos de noticias sospechosas anteriormente mencionados.

Así pues, partiendo de un conjunto de muestras de publicaciones de Twitter con ambos tipos de noticias, se llevó a cabo la clasificación de estas en noticas reales o sospechosas para, posteriormente, analizar las diferencias entre ambos tipos de publicaciones haciendo uso para ello de un análisis estadístico. De esta manera se determinó que ambos tipos de contenido presentaban diferencias muy significativas debido a la subjetividad empleada en el lenguaje y al uso de fundamentos morales, pues las noticias sospechosas tendían a narrar en lugar de simplemente limitarse a informar de un hecho o acontecimiento como en el caso de las noticias reales.

- En cuarto lugar, cabe destacar otra propuesta la cual consistía en estudiar y demostrar la influencia que pueden ejercer los bots (cuentas controladas por software) en la difusión de noticias falsas a través de las redes sociales. Para ello, se propone [27]:
    - **Analizar la probabilidad de que una cuenta que ha publicado un contenido concreto sea un bot.**

      Inicialmente en este estudio se rastrearon artículos de Twitter publicados por organizaciones independientes de verificación de datos (hechos verídicos) y por páginas web dedicadas a la publicación de noticias falsas o engañosas (hechos no demostrados). Así pues, estas publicaciones junto a todos sus metadatos fueron recopiladas usando la API de Twitter.

      Para determinar si las cuentas ligadas a las publicaciones a estudiar eran bots se utilizó "Botometer". Se trata de una herramienta encargada de ponderar a los usuarios asociándoles una nota de "bot", la cual indica el grado de cercanía que tiene ese usuario con una cuenta controlada por software. Esta solución se incorporó en un algoritmo de aprendizaje automático entrenado con miles de ejemplos de cuentas reales y cuentas controladas por bots.

      Con el transcurso de las semanas, se observó que las publicaciones con contenido falso en su mayoría pasaban desapercibidas, mientras que un grupo reducido de ellas se volvía viral. Analizando publicaciones virales en sus distintas fases de propagación, se determinó que esta difusión desmesurada comenzaba al compartir dicho contenido sin cesar con grandes cantidades de usuarios.
    - **Analizar la influencia que ejerce un determinado usuario autor de una publicación** en base al número de seguidores que tiene.

      De esta manera, analizando la propagación de noticias falsas se observó que los bots difundían estas publicaciones o enlaces focalizando especialmente en usuarios que ejercieran una fuerte influencia en la red social (famosos, periodistas, políticos, …) presentando una importante cantidad de seguidores. De esta manera, los bots conseguían que esas noticias falsas adoptaran la mayor visibilidad y difusión posible. Además, se tuvo en cuenta la localización geográfica que presentaban estos bots y se observó que el patrón de conducta que seguían no coincidía con sus supuestas ubicaciones a nivel de ámbito político y/o social.

De esta manera, se comprobó que los bots afectaban e influían sobre la opinión de los usuarios presentes en la red, pues la mayoría de retweets que habían recibido las

publicaciones con contenido falso eran de personas reales, demostrando la vulnerabilidad que sufren muchos usuarios lectores. Por tanto, los bots resultan ser una herramienta eficaz para manipular la información circulante en los medios sociales y así engañar o desinformar a los usuarios. Ello llevó a pensar que una buena medida para frenar esta propagación de noticias falsas sería comenzar frenando a este tipo de cuentas controladas por software.

3.4. Credibilidad de una cuenta

Hoy en día reconocer una cuenta de un usuario real y veraz resulta un trabajo complejo ya que, como se ha mencionado con anterioridad, en la actualidad es común que existan cuentas de usuario falsas o manejadas por bots y que pueden haber sido creadas por motivos diversos como la difusión de contenido fraudulento o con fines publicitarios entre otros.

Así pues, el hecho de poder identificar y catalogar la fiabilidad de una cuenta resulta ser un hecho relevante a estudiar y un factor determinante a la hora de otorgar mayor o menor credibilidad a los contenidos publicados en las redes sociales. Determinar qué factores caracterizan la credibilidad de una cuenta abarca un abanico muy amplio de opciones y, por tanto, existen diversas propuestas de indicadores según distintos estudios realizados:

Una **propuesta realizada por la universidad americana Carnegie Mellon y la empresa Microsoft**, consiste en **evaluar la** credibilidad de las cuentas de usuario de Twitter s**egún su foto, nombre y ortografía** [35]. El estudio demostraba la incredulidad de los usuarios ante aquellos perfiles que:

- No presentaban fotografía propia y que en su lugar presentaban algún tipo de avatar o icono como foto de perfil.
- Tenían nombres de cuenta ficticios o apodos que, claramente, no eran nombres propios.
- Presentaban un mayor número de usuarios seguidos que de seguidores.
- Presentaban publicaciones cuyo contenido tuviera faltas de ortografía o gramática.

Otra **propuesta realizada por la universidad de Clarckson de Estados Unidos** consiste en, **dado un conjunto de datos de perfiles** de Twitter de carácter público, **analizar detenidamente los atributos clave de cada perfil** con el fin de **identificar perfiles falsos** [37]. Para ello se tomaron como indicadores clave a analizar los siguientes atributos:

- Identificador de la cuenta
- Número de seguidores

- Número de cuentas seguidas
- Si es una cuenta verificada o no
- Fecha y hora de creación de la cuenta
- Descripción
- Localización geográfica
- Fecha y hora de actualización de la cuenta
- URL de la imagen de perfil
- Nombre de pantalla / nombre de usuario
- Historial de publicaciones o tweets

Con este conjunto de atributos se pretendía pues identificar un conjunto de perfiles falsos que fuera confiable y sobre el cual poder posteriormente analizar los distintos patrones de similitud existentes entre este tipo de perfiles.

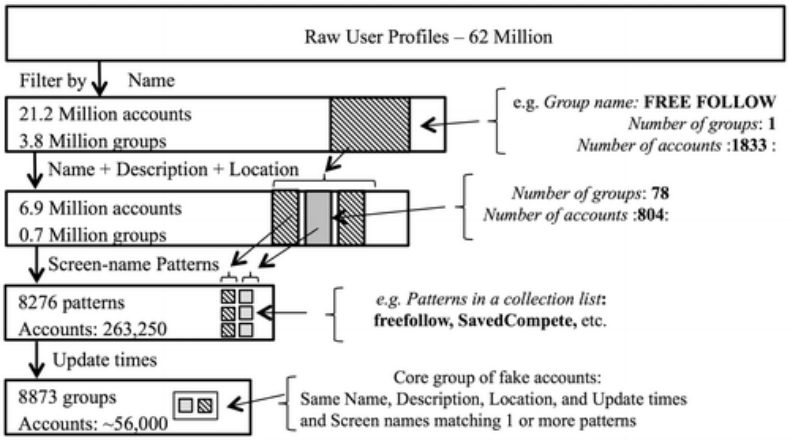


Figura 7. Diagrama esquemático sobre el algoritmo de identificación de grupos falsos de perfiles.
(Fuente https://journals.sagepub.com/)

Una vez filtradas las cuentas con alta probabilidad de ser falsas, se analizó cada uno de los atributos clave que presentaban dichas cuentas con el fin de identificar y/o detectar si existían patrones de similitud entre dichas cuentas falsas confiables. Tras ello, el estudio determinó las siguientes conclusiones:

- La gran mayoría de los atributos clave de perfil coincidían entre varias cuentas (conjuntos de perfiles similares).
- Utilizando la entropía de Shannon [36] sobre los nombres de los perfiles se determinó que muchos de ellos habían sido creados en masa, es decir, presentaban nombres prácticamente idénticos, pero con alguna variante (Ejemplo: name1, name2, name3, …).

- Se observó que las cuentas falsas tendían a seguir a otros usuarios de una manera más rápida con el paso del tiempo, mientras que en cuanto al número de seguidores las cuentas falsas acumulan menores cantidades que las cuentas reales.

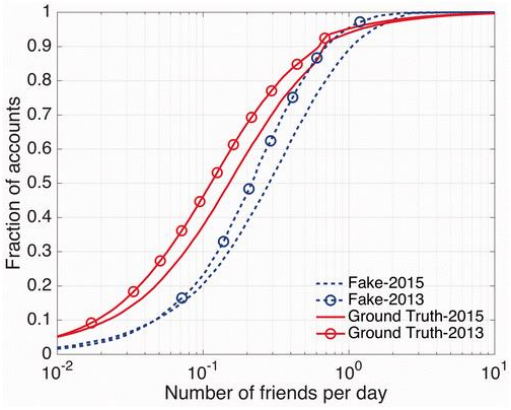


Figura 8. Distribución del número de perfiles seguidos normalizado para cuentas reales y ficticias en los años 2013 y 2015.
(Fuente https://journals.sagepub.com/)

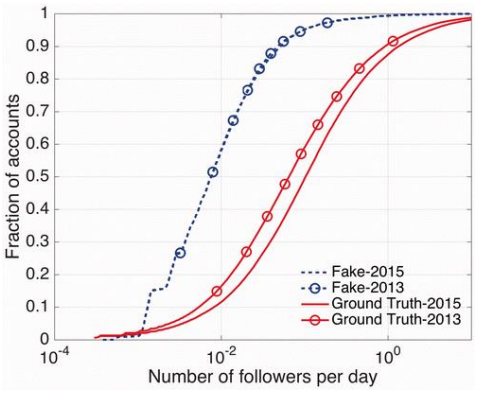


Figura 9. Distribución del número de seguidores normalizado para cuentas reales y ficticias en los años 2013 y 2015.
(Fuente https://journals.sagepub.com/)

- Si se analizaba el historial de publicaciones de las cuentas, se observaba que varias de ellas habían compartido en numerosas ocasiones contenidos idénticos para una misma fecha y hora.
- Analizando los tiempos de actualización y/o creación de cuenta se observó que determinados grupos de perfiles muy similares entre ellos presentaban tiempos de actualización y/o creación idénticos o con una diferencia mínima de escasos segundos.

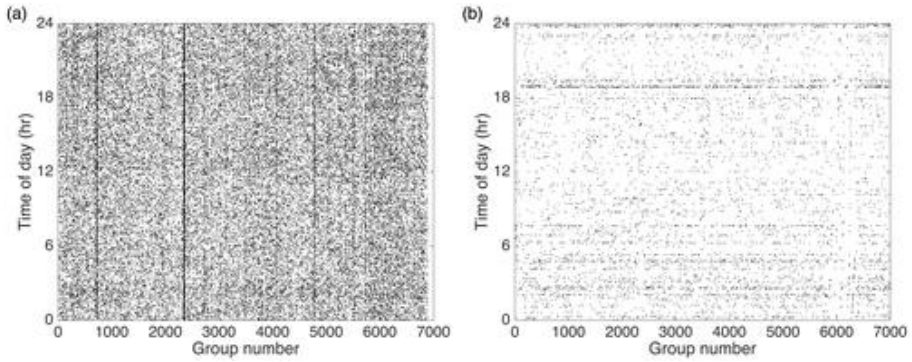


Figura 10. Comparación entre tiempos de actualización de perfiles reales (a) y perfiles falsos (b). (Fuente https://journals.sagepub.com/)

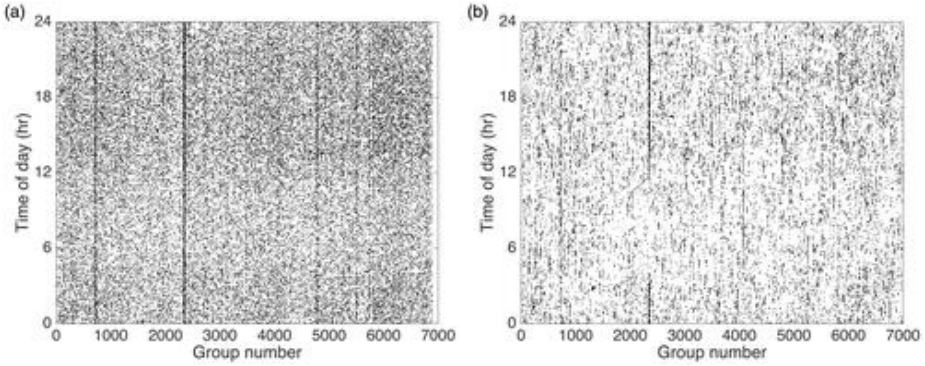


Figura 11. Comparación entre tiempos de creación de perfiles reales (a) y perfiles falsos (b). (Fuente https://journals.sagepub.com/)

3.5. Botometer

Botometer es un algoritmo desarrollado por el IUNI (Network Science Institute) y el CNetS (Complex Networks and Systems Research) en la Universidad de Indiana. Se encarga de controlar y/o verificar la actividad de una cuenta de Twitter especificada y de ponderar a este usuario en base al grado de similitud que presente con un bot [28].

Este algoritmo se basa en miles de ejemplos y, mediante el uso de la API de Twitter, obtiene el perfil público de la cuenta indicada, así como gran cantidad de sus metadatos (tweets, menciones, amigos, …). Una vez obtenida toda esta información, caracteriza la cuenta utilizando para ello varios modelos de aprendizaje automático con el fin de otorgarle una puntuación final. Esta clasificación diferencia principalmente entre características específicas para cuentas en inglés y características generales independientes del lenguaje. Así pues, las puntuaciones obtenidas oscilarán entre el 0 y el 5 (representadas en distintos colores), siendo el valor más bajo el que indique cuando se trata de un usuario real [29].

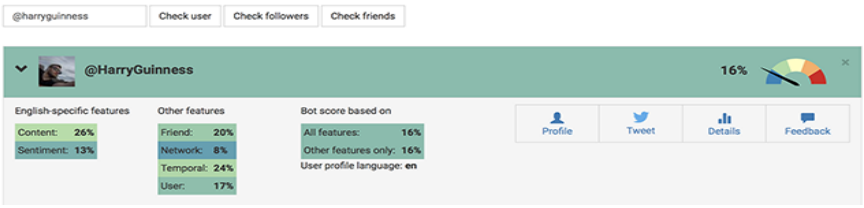


Figura 12. Ejemplo de posible persona detectada con Botometer.
(Fuente https://www.howtogeek.com/)

De esta manera, cuanta más puntuación se obtenga mayor será la probabilidad de que esa cuenta esté siendo controlada por algún tipo de software.

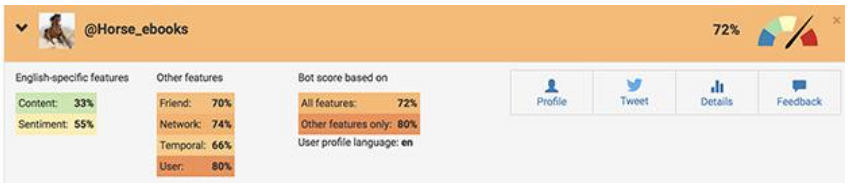


Figura 13. Ejemplo de posible bot detectado con Botometer.
(Fuente https://www.howtogeek.com/)

Botometer está disponible a través de una API pública (botometer.iuni.iu.edu) y para poder usar la herramienta será necesario iniciar sesión en una cuenta de Twitter y otorgar los permisos de lectura necesarios. Además, su página web principal pone a disposición las siguientes librerías cliente oficiales [30]:

- Librería API HTTP en su versión gratuita y de pago.
- Librería cliente para Python compatible con Python 2.7 y 3.4 o superior.

3.6. API de Twitter

Twitter otorga la posibilidad de compartir sus datos tanto con empresas como con desarrolladores y/o usuarios particulares. Para ello, tal y como se indica dentro de su apartado de reglas y políticas [7], ofrece sus interfaces de programación de aplicaciones o API. Mediante estas herramientas, permite el acceso a datos de usuarios con perfiles públicos en la red social de manera predeterminada y/o gestionar cuentas de usuario (propias o ajenas autorizadas) con contenido privado haciendo uso de otra API compatible. Este acceso al contenido no se otorga de manera deliberada, pues Twitter ha de autorizar previamente a todos aquellos usuarios y organismos que soliciten acceso a su contenedor de datos. Con todo ello, Twitter pretende facilitar el desarrollo de aplicaciones capaces de integrarse con la red social.

Así pues, para poder acceder a cualquiera de las API de Twitter, es necesario crear una cuenta de desarrollador (“developer”), mediante la cual se deberá registrar la aplicación con la que se

pretende hacer uso de los datos. Tal y como se menciona anteriormente, por defecto Twitter solo permite extraer datos de carácter público. Sin embargo, ofrece la posibilidad de otorgar acceso a puntos de conexión concretos de ámbito privado, que requerirán permisos adicionales por parte del usuario propietario de la cuenta. Por ello, el usuario podrá gestionar desde sus preferencias de cuenta los permisos que otorga sobre el uso de sus datos privados a cada aplicación.

Así pues, Twitter clasifica los puntos de conexión que ofrece en cinco grupos:

- **Cuentas y usuarios:** ofrece la posibilidad de gestionar cuentas de usuario a desarrolladores autorizados.
- **Tweets y respuestas:** ofrece acceso a las publicaciones de carácter público y permite realizar publicaciones. Estos puntos de conexión se usan en muchos casos para identificar el origen de rumores y la propagación de estos.
- **Mensajes directos:** ofrece acceso a los mensajes privados del usuario a las aplicaciones que éste haya autorizado previamente con el fin de crear experiencias personalizadas para los usuarios.
- **Anuncios:** ofrece un conjunto de API con los que facilitar la realización de campañas publicitarias a empresas permitiendo identificar los intereses de los usuarios y tendencias seguidas por los mismos.
- **Herramientas y SDK del editor:** ofrece herramientas para desarrolladores de software con las que poder integrar elementos principales de Twitter en páginas web como por ejemplo los botones para seguir a un usuario o compartir un determinado contenido.

3.6.1. Acceso a la API: Autenticación

Para poder llevar a cabo las peticiones de información sobre cualquier tipo de API de Twitter, es necesario utilizar autenticación OAuth o autenticación básica.

Tal y como se indica en la documentación oficial de la API de Twitter [8], dependiendo del tipo de API a la que se pretende acceder y el tipo de información que se pretende extraer, será necesario un método de autentificación u otro.

Tipo de API:	Método de autentificación:
API de REST y streaming estándar	Oauth2 (token de portador), Oauth 1a (usuario de la aplicación)
API premium	Oauth2 (token de portador), Oauth 1a (usuario de la aplicación)
API de empresa	Autenticación básica, Oauth2 (token de portador), Oauth 1a (usuario de la aplicación)
API de anuncios	Oauth 1a (usuario de la aplicación)

Figura 14. Autentificaciones según el tipo de API
(Fuente https://developer.twitter.com/)

En primer lugar, la autenticación básica, tal y como se indica en la documentación oficial de la API de Twitter [9], está destinada a las API de tipo empresarial de Twitter y consiste en autentificarse mediante un correo electrónico y una contraseña válidos, también conocido como autenticación básica HTTP. Ambos datos de identificación se transmiten en la cabecera de las solicitudes.

Por otra parte, la autenticación OAuth, tal y como su propia web oficial indica [10], resulta ser un protocolo de delegación utilizado para otorgar permisos a una red de aplicaciones y API concretas. OAuth no necesita conocer ningún dato del usuario que pretende autenticarse para acceder al contenido solicitado, únicamente requerirá que se proporcione un "token" o enlace de solicitud para permitir el acceso.

El tipo de API que es necesaria en este proyecto es "API de REST y streaming estándar", ya que se pretende extraer datos para su posterior análisis. Usando dicho tipo de API, podrá hacerse uso de dos métodos distintos de autenticación OAuth: OAuth 2 y OAuth 1a [11]. Así pues, estos métodos consisten en:

- **OAuth 2**: el método de autenticación OAuth2, representa la autenticación solo de la aplicación y se encarga de permitir que una aplicación actúe en su propio nombre sin involucrar al usuario como tal.

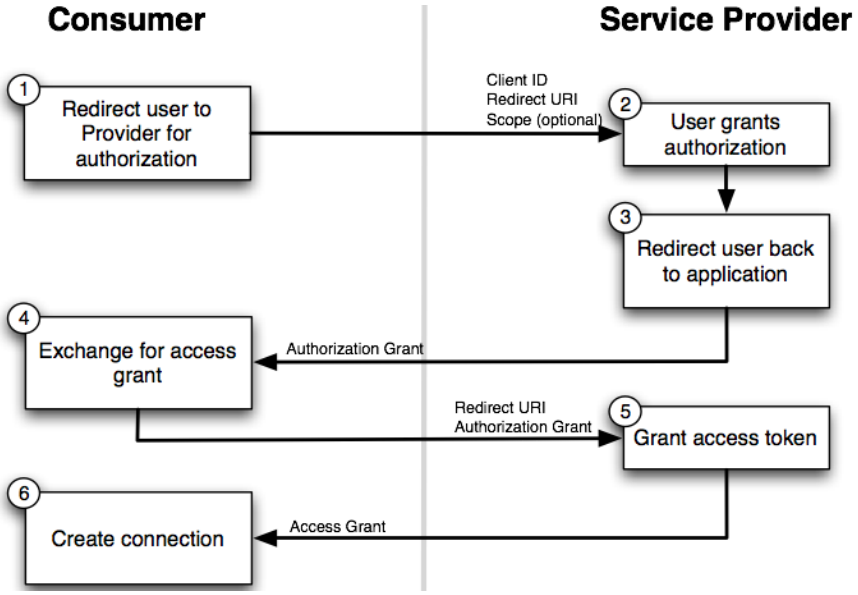


Figura 15. Esquema sobre el funcionamiento de OAuth 2
(Fuente https://stackoverflow.com/)

Esta autenticación es utilizada principalmente por desarrolladores que necesitan acceder a recursos de carácter público. De esta manera, OAuth 2 para funcionar requiere de:

- **Clave de consumidor**
- **Clave de consumidor secreta**

- **OAuth 1a**: el método de autenticación OAuth 1a, representa la autenticación del usuario de la aplicación y se encarga de permitir que una aplicación autorizada con OAuth 2, actúe en nombre de y como si fuera el propio usuario en cuestión. De esta manera, una petición certificada identifica la aplicación que actúa contra la API en nombre del usuario, así como los permisos que le han sido otorgados.

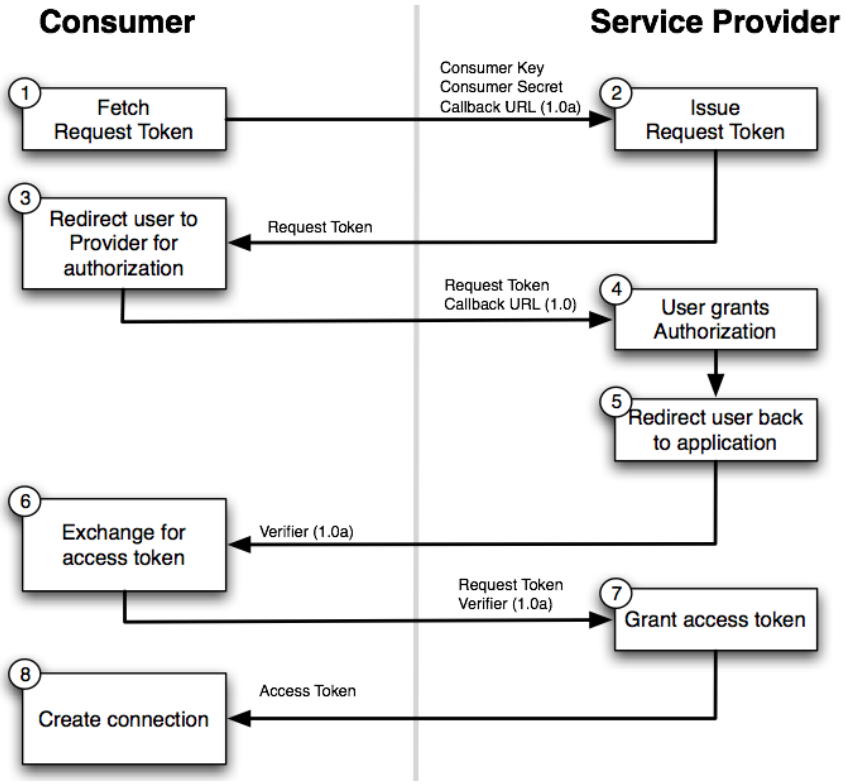


Figura 16. Esquema sobre el funcionamiento de OAuth 1
(Fuente https://stackoverflow.com/)

Esta autenticación es utilizada principalmente para casos en los que se requiera extraer información específica del usuario o para actuar en su nombre. Así pues, OAuth 1a para funcionar requiere de cuatro elementos que provienen de la aplicación de Twitter creada por el usuario desarrollador:

- **Clave de consumidor**
- **Clave de consumidor secreta**
- **"Token" de acceso**
- **"Token" de acceso secreto**

Todas las claves y "tokens" requeridos para ambos métodos de autenticación, serán proporcionados por la plataforma de Twitter para desarrolladores una vez se haya registrado la aplicación con la cual se vaya a acceder a la API.

3.7. Python

Python es un lenguaje de programación orientado a objetos e interpretado, el cual presenta una semántica dinámica que facilita el desarrollo de páginas web y aplicaciones. Además, presenta

una sintaxis sencilla e intuitiva, así como numerosas librerías, distribución libre y compatibilidad en múltiples plataformas, hechos que favorecen que muchos desarrolladores opten por este lenguaje a la hora de afrontar un proyecto.

En los últimos años, el lenguaje Python se ha convertido en uno de los más populares y utilizados según el índice TIOBE, tal y como se muestra en la **¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.**. Uno de los detalles que le hacen destacar enormemente resulta ser el uso de "machine learning" en algunos de sus paquetes con el fin de procesar grandes volúmenes de datos de manera precisa y robusta [12].

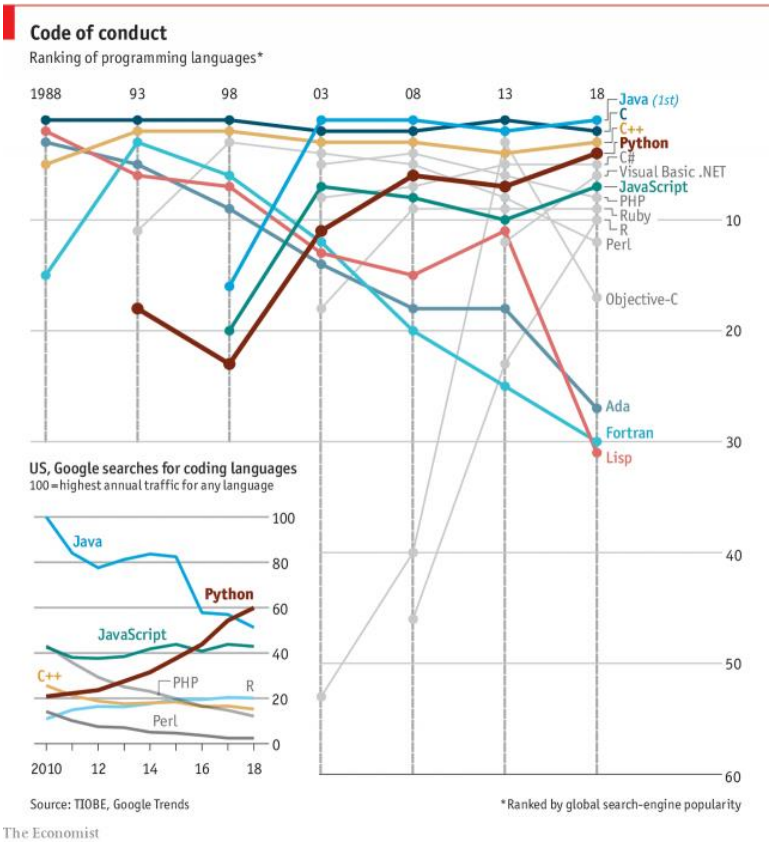


Figura 17. Ranking TIOBE de lenguajes de programación (2018)
(Fuente https://www.economist.com/)

Su popularidad no cesa de crecer entre la comunidad de desarrolladores y es que, según el índice PYPL, basándose en el análisis de las búsquedas de Google sobre tutoriales de lenguajes de programación para el mes de enero de 2019, las búsquedas sobre Python han supuesto un cuarto del total registrado [13].

Además, como se ha mencionado anteriormente, resulta ser uno de los lenguajes con mayor número de librerías destinadas a la plataforma de Twitter, tal y como se indica en la documentación oficial para desarrolladores de la API [14].

De entre estas librerías cabe destacar 3 en concreto, las cuales permiten la extracción de datos de carácter público de la red social, funcionando a modo de contenedores de información:

- **Tweepy**
- **Twython**
- **TwitterAPI**

Para acceder a la API de la red social Twitter y llevar a cabo dicha extracción, será necesario previamente realizar la correspondiente autenticación con el método OAuth2 ya que se pretende recoger información pública y únicamente será necesario dar autorización a la aplicación, utilizando para ello las distintas clases y métodos que recogen las anteriores librerías citadas.

3.7.1. Librería Tweepy

Para llevar a cabo la tarea de autenticación OAuth, la librería Tweepy utiliza la clase "tweepy.AuthHandler", a la cual se le deberá pasar como parámetro la clave de consumidor y la clave de consumidor secreta [15]. Además, la clase principal de la librería, "tweepy.API", se encarga de establecer la conexión con la API de Twitter mediante la autenticación anterior pasada como parámetro y de almacenar los datos finalmente proporcionados por la API.

Así pues, se procede a listar algunos de los métodos de la clase "tweepy.API" con mayor relevancia en la extracción concreta de la información que necesitemos del contenedor [16]:

- **tweepy.API.user_timeline()**: Devuelve por defecto los últimos 20 "retweets" y/o tweets publicados por el usuario pasado como parámetro. En el caso de querer otra cantidad de estados se debe especificar con parámetro adicional.
- **tweepy.API.get_status():** Devuelve información variada sobre el tweet pasado como parámetro (contenido, geolocalización, fecha de creación, autor, …).
- **tweepy.API.get_user():** Devuelve información variada sobre el usuario pasado como parámetro (identificador del usuario, colores de su perfil, si tiene o no fondo de perfil, número de seguidores, número de favoritos, …).
- **Tweepy.API.retweets():** Devuelve por defecto los 100 primeros "retweets" de la publicación pasada como parámetro. En el caso de querer otra cantidad se debe especificar con otro parámetro adicional.
- **tweepy.API.followers():** Devuelve los seguidores del usuario pasado como parámetro o, en su defecto, del usuario autentificado.

- **tweepy.API.favorites():** Devuelve los favoritos del usuario pasado como parámetro o, en su defecto, del usuario autentificado.
- **tweepy.API.friends():** Devuelve las cuentas que sigue el usuario pasado como parámetro o, en su defecto, el usuario autentificado.
- **tweepy.API.exists_friendship():** Dados como parámetros un usuario A y un usuario B, devuelve verdadero en el caso de que el usuario A siga al usuario B y falso en caso contrario.
- **tweepy.API.search():** Realiza una búsqueda de "tweets" cuyo contenido coincida con la consulta pasada como parámetro.
- **tweepy.API.search_users():** Realiza una búsqueda de usuarios en la red social cuyo nombre coincida con la consulta pasada como parámetro. Está limitado a los 1000 primeros resultados.

3.7.2. Librería Twython

La clase principal de la librería Twython resulta ser "twython.Twython", la cual es utilizada para llevar a cabo la tarea de autenticación OAuth [17], pasando como parámetro la clave de consumidor y la clave de consumidor secreta con el fin de efectuar la conexión con la API de Twitter. Además, dicha clase también se utiliza para almacenar los datos sustraídos de la API en una especie de contenedor.

Así pues, los métodos más relevantes que presenta esta clase para la sustracción de información concreta y/o filtrada del contenedor de datos, resultan ser los siguientes [18**¡Error! No se encuentra el origen de la referencia.**]:

- **twython.Twython.get_user_timeline():** Devuelve una colección con los "retweets" y los "tweets" más recientes del usuario pasado como parámetro.
- **twython.Twython.show_status():** Devuelve información variada sobre el "tweet" especificado como parámetro (contenido, fecha de creación, autor, …).
- **twython.Twython.show_user():** Devuelve información variada sobre el usuarios especificado como parámetro (identificador del usuario, colores de su perfil, si tiene o no fondo de perfil, número de seguidores, número de favoritos, …).
- **twython.Twython.get_retweets():** Devuelve por defecto los 100 primeros "retweets" de la publicación especificada como parámetro.
- **twython.Twython.get_favorites():** Devuelve por defecto los últimos 20 "tweets" que el usuario especificado ha marcado con favorito. En el caso de querer otra cantidad se debe especificar como parámetro.

- **twython.Twython.get_followers_list():** Devuelve una colección con los usuarios que siguen al usuario pasado como parámetro.
- **twython.Twython.get_friends_list():** Devuelve una colección con los usuarios a los que sigue el usuario especificado como parámetro.
- **twython.Twython.search():** Devuelve una colección de "tweets" cuyo contenido coincida con la consulta indicada como parámetro.
- **Twython.Twython.search_users():** Realiza una búsqueda de usuarios cuyos nombres coincidan con la consulta pasada como parámetro.

3.7.3. Librería TwitterAPI

La librería TwitterAPI presenta como clase principal "TwitterAPI", sobre la cual se deberán indicar como parámetros la clave de consumidor y la clave de consumidor secreta para llevar a cabo la autenticación OAuth sobre la API de Twitter [19]. Además, esta clase resulta ser la principal de la librería y, junto con el método principal "TwitterAPI.request" extrae distintos datos procedentes de la API de Twitter. Se trata de peticiones HTTP POST y se deberá indicar qué datos queremos extraer como parámetro, siendo las siguientes peticiones algunas de las más relevantes [20]:

- **TwitterAPI.request('statuses/show/:%d' % <ID del Tweet>):** esta petición devuelve información variada sobre el tweet especificado con su ID o identificador (contenido de la publicación, fecha y hora de creación, número de favoritos que ha recibido, …).
- **TwitterAPI.request('search/tweets',{'q':'<palabra a buscar>'}):** esta petición devuelve un grupo de "tweets" en cuyo contenido aparece la palabra indicada como parámetro.
- **TwitterAPI.request('statuses/filter', {'locations':'<Coordenadas de la localización>'}):** permite filtrar contenido publicado que proceda de la localización deseada, indicando sus coordenadas como parámetro.
- **TwitterAPI.request('statuses/filter', {'track':<Palabra o palabras para filtrar}):** permite filtrar contenido que contenga la palabra o palabras especificados como parámetro.

3.7.4. Comparativa

La librería Tweepy resulta ser la más famosa y reconocida por la comunidad de desarrolladores, presentando una documentación muy completa y una fuerte presencia en la red, pudiendo encontrar importantes cantidades de tutoriales sobre su uso. Además, resulta fácil e intuitiva de manejar y la gran variedad de métodos que presenta permite abarcar un amplio abanico de opciones a usar sobre la API de Twitter.

Por otra parte, la librería Twython presenta métodos muy similares a la anterior. Sin embargo, tiene algunas limitaciones tales como menor cantidad de métodos destinados a la extracción de contenido público de Twitter y/o menor cantidad de documentación disponible sobre ella en la red. Todo ello ha repercutido enormemente en que no sea tan bien valorada por la comunidad de desarrolladores como lo es Tweepy.

Por otro lado, la librería TwitterAPI presenta una documentación muy escueta e insuficiente. Sus sentencias para realizar solicitudes de información no resultan nada intuitivas ni rápidas, hecho que dificulta enormemente su uso y aplicación. Por ello, los desarrolladores suelen optar por alguna de las dos opciones anteriormente comentadas, pues presentan mayor cantidad de documentación en línea, así como una mayor versatilidad debido a la importante cantidad de métodos que podemos encontrar en sus respectivas clases principales.

Tabla 2. Comparativa entre las librerías de Python

	Tweepy	Twython	TwitterAPI
Documentación	Abundante	Normal	Escasa
Dificultad	Baja	Baja	Alta
Popularidad	Alta	Normal	Baja
Cantidad de funcionalidades disponibles	Abundante	Normal	Escasa

4. Objetivos

El objetivo de este proyecto consiste en el desarrollo de una herramienta con la que poder determinar si una noticia es más o menos fiable en base al estudio en el tiempo de la credibilidad de su autor, con el fin de colaborar para frenar la desinformación existente en la red.

Gracias a la plataforma de Twitter, se dispone de un escenario verdadero que poder analizar, repleto de contenido público y que cambia constantemente en tiempo real. Así pues, recolectando información de este gran conjunto de datos, la tendencia del **grado de credibilidad que puede tener un usuario en Twitter** se podrá estudiar y valorar con mayor profundidad y exactitud gracias a la gran cantidad de metadatos que se podrán obtener de la plataforma.

Uno de los aspectos clave para probar la fiabilidad de una cuenta resulta ser el análisis de sus características, es decir, la cantidad de seguidores que tiene, usuarios a los que sigue, cantidad de publicaciones o la actividad que presenta dicha cuenta entre otros aspectos. Además, verificar si una cuenta pertenece a una persona real o está siendo manejada por un bot (cuenta controlada por software) será de especial relevancia a la hora de poder atribuirle un mayor o menor grado de credibilidad.

Se pretende por tanto con este proyecto analizar en el tiempo la credibilidad de las cuentas de usuario de la red social Twitter, estableciendo para ello una serie de indicadores los cuales serán ponderados con una puntuación que variará en función de su contribución en mayor o menor medida a qué la cuenta sea considerada como fiable. Además, la posible variación de esta puntuación en el tiempo será de especial relevancia a la hora de estudiar la tendencia de la credibilidad de una cuenta de usuario. Así pues, el grado de credibilidad de una cuenta resultará ser pues la suma total de los indicadores establecidos: cuanta mayor sea la puntuación final obtenida, mayor fiabilidad presentará dicha cuenta de usuario.

Esta herramienta será aplicada a un determinado número de cuentas de las cuales se conoce con antelación su resultado, con el fin de probar su efectividad y extraer conclusiones sobre los resultados obtenidos a lo largo del tiempo. Este conjunto sujeto a estudio estará formado por dos grupos principales: un primer grupo formado por cuentas que se tiene constancia de que pertenecen a personas u organizaciones reales y, por otro lado, un segundo grupo conformado por cuentas de las que se sabe que son falsas, poco fiables o se sospecha que están siendo controladas por bots. Así pues, el estudio de la credibilidad que presente un usuario y su

evolución a lo largo del tiempo será un hecho determinante para valorar la fiabilidad de su contenido.

5. Propuesta y validación

Para determinar la credibilidad de una cuenta es necesario definir y/o establecer aquellos indicadores que van a ser necesarios para forjar la puntuación final asociada al usuario. Estos indicadores se nutren de características de la propia cuenta que habrá que extraer previamente de la API de Twitter. Sin embargo, todos ellos no serán igual de relevantes, pues unos serán más concluyentes y/o decisivos (factores determinantes) que otros (factores no determinantes) a la hora de obtener la puntuación final. De esta manera, cuanto mayor sea la puntuación final asociada a la cuenta de un usuario más fiable resultará y, por el contrario, cuanto menor sea la puntuación final menos creíble será dicho perfil.

Así pues, para calcular l**a puntuación de credibilidad de una cuenta de Twitter** ($\boldsymbol{P(CC_t)}$) se propone la siguiente fórmula:

$$P(CC_t) = |1 + \sum_{i=1}^{n}(P(F_{nd}))| \cdot \prod_{j=1}^{m}(P(F_d))$$

Donde:

$\{P(F_{nd_1}), P(F_{nd_2}), \dots (F_{nd_n})\}$ – Puntuaciones de los factores no determinantes.

$\{P(F_{d_1}), P(F_{d_2}), \dots (F_{d_m})\}$ – Puntuaciones de los factores determinantes.

Tal y como se puede observar en la fórmula, la puntuación de la credibilidad de una cuenta de Twitter vendrá dada por el sumatorio de un conjunto de factores no determinantes que, posteriormente, se multiplicará por los factores determinantes.

De esta forma, si desglosamos la fórmula anterior en todos los factores que la componen quedaría de la siguiente manera:

$$P(CC_t) = |\,1 + ((P(R_d) + P(T_d) + P(S_u) + P(R_{ss}) + P(A_c) + P(D_p) + P(L_p) + P(N_l) + P(T_a) + P(P_p) + P(P_b))| \cdot (\,P(V_c) \times P(I_p)\,)$$

Donde:

<u>Factores no determinantes (F_{nd})</u>

$P(R_d) -$ Puntuación según los retweets diarios realizados por el usuario.

$P(T_d)$ – Puntuación según la cantidad de tweets diarios publicados por el usuario.

$P(S_u)$ – Puntuación según la cantidad de seguidores que tiene el usuario.

$P(R_{ss})$ – Puntuación según el ratio seguidores/seguidos.

$P(A_c)$ – Puntuación según la antigüedad de la cuenta.

$P(D_p)$ – Puntuación según si el usuario dispone de una descripción en su perfil o no.

$P(L_p)$ – Puntuación según si el usuario ha compartido su localización geográfica o no.

$P(N_l)$ – Puntuación según la cantidad de "likes" dados por el usuario.

$P(T_a)$ – Puntuación según la media de publicaciones en base a la antigüedad de la cuenta en días.

$P(P_p)$ – Puntuación según si el usuario ha personalizado el diseño de su perfil o no.

$P(P_b)$ – Puntuación según el resultado obtenido para el usuario en Botometer.

Factores determinantes (F_d)

$P(V_c)$ – Puntuación según si la cuenta está verificada o no.

$P(I_p)$ – Puntuación según si la cuenta tiene foto de perfil o no.

Cada una de las puntuaciones indicadas anteriormente se calcula de forma independiente y de manera distinta.

De esta puntuación final, se pretende estudiar su tendencia a lo largo del tiempo. Así pues, se distinguen tres escenarios posibles para la tendencia:

- Aquellos usuarios reales tendrán más o menos una tendencia constante con poco margen de cambio, es decir, lo más plana posible. Sin embargo, si se obtiene una tendencia plana con valores muy bajos tampoco resultaría válido, es decir, es necesario establecer un valor mínimo de calidad que deberá presentar la puntuación final $P(CC_t)$. Así pues, se tomará como valor mínimo de calidad que la puntuación final deba alcanzar al menos los 100 puntos.
- Por otra parte, se podría obtener una tendencia negativa o muy negativa, hecho que puede indicar que la cuenta estudiada podría estar dejando de utilizarse o, directamente, el usuario la ha abandonado.

- Otra opción posible, sería obtener una tendencia positiva o muy positiva. Así pues, obtener una tendencia muy positiva indica una circunstancia extraña donde posiblemente sea algo derivado de campañas de publicidad, tácticas profesionales o comerciales o algún tipo de manipulación similar. Sin embargo, una tendencia ligeramente positiva entra dentro del crecimiento normal que puede experimentar una cuenta de usuario.

Mientras que el cálculo de las puntuaciones de los diferentes factores determinantes y no determinantes y el cálculo de la puntuación final son automatizables, la clasificación de la cuenta en base al comportamiento de la tendencia de su $P(CC_t)$ a lo largo del tiempo va a ser completamente especular.

5.1. Factores no determinantes

Para obtener las **puntuaciones de los factores no determinantes** se parte de una puntuación base para cada indicador. En el caso de que el aspecto que está siendo evaluado no contribuya positivamente con la credibilidad del usuario, se le irán restando puntos, pero nunca podrá llegar a un valor menor que 1 la puntuación referente a los factores no determinantes. Por ello, se sumará 1 inicialmente y se efectuará el valor absoluto a este bloque.

5.1.1. Puntuación según los retweets diarios realizados por el usuario

Tal y como se ha mencionado en apartados anteriores, es frecuente encontrar cuentas de Twitter que están siendo manejadas por algún tipo de software (bots) o simplemente se trate de cuentas falsas manejadas por un tercero. Este tipo de cuentas se crean con intenciones de difundir determinados contenidos, pudiendo darse esta propagación a través de retweets o tweets. El número de retweets es un factor importante ya que, en el caso de este tipo de cuentas fraudulentas, algunos comportamientos típicos de estos perfiles resultan ser retuitear deliberadamente determinado contenido, como es el caso de muchas campañas de marketing donde empresas o individuos pueden contratar este servicio [43]. Otro comportamiento alternativo que puede adoptar una cuenta poco creíble es no retuitear, es decir, en el caso de las cuentas falsas dedicadas a publicar contenido masivamente, su finalidad no es retuitear publicaciones pues se centran en realizar ellas mismas en tuitear contenidos. Normalmente, de forma diaria, una cuenta real suele hacer algún que otro retweet a lo largo del día y, según algunos expertos, si los usuarios desean aumentar su repercusión en la red social se recomienda realizar aproximadamente 7 retweets de manera diaria [42]. Por ello, se distinguen dos comportamientos sospechosos en relación con el número de retweets dados por día: ausencia

de retweets (cantidad igual a 0) y cantidad alta de retweets (en este caso se tomará como valor alto que la cantidad sea mayor que 10). Así pues, para calcular este indicador se ha decidido partir de una puntuación inicial de 100 puntos a la cual le serán descontados 50 puntos en el caso de que la cantidad de retweets diaria supere las 20 y, en el caso de que no se realice ningún retweet, se descontarán 100 puntos.

$$P(R_d) = 100, \textit{ cuando } 0 < \textit{cantidad\_retweets} \leq 10$$
$$P(R_d) = 100 - 50 = 50, \textit{ cuando cantidad\_retweets} > 10$$
$$P(R_d) = 100 - 100 = 0, \textit{ cuando cantidad\_retweets} = 0$$

5.1.2. Puntuación según la cantidad de tweets diarios publicados por el usuario

Al igual que en el caso de los retweets, los tweets pueden ser de igual forma para muchas empresas o asociaciones una vía mediante la cual publicar contenido de manera masiva para publicitarse [43]. Los expertos recomiendan realizar entre 10 y 20 publicaciones diarias para obtener una mayor repercusión e importancia en la red social [44]. Un usuario real en la plataforma, que esté más o menos activo, realiza alguna que otra interacción diaria, es decir, resulta raro que no presente ningún tipo de interacción diaria en su perfil. Contrariamente, otro hecho que puede sembrar duda sobre el usuario es cuando la cuenta pública masivamente contenido de forma diaria. Así pues, se distinguen para este estudio dos comportamientos sospechosos en relación con el número de tweets publicados al día: ausencia de publicaciones (cantidad igual a 0) y cantidad alta de tweets (en este caso se tomará como valor alto que la cantidad sea mayor que 20). Así pues, para calcular este indicador se ha decidido partir de una puntuación inicial de 100 puntos a la cual le serán descontados 50 puntos en el caso de que la cantidad de publicaciones diarias supere los 20 y, en el caso de no publicar ningún tweet, se descontarán 100 puntos.

$$P(T_d) = 100, \textit{ cuando } 0 < \textit{cantidad\_tweets} \leq 20$$
$$P(T_d) = 100 - 50 = 50, \textit{ cuando cantidad\_tweets} > 20$$
$$P(T_d) = 100 - 100 = 0, \textit{ cuando cantidad\_tweets} = 0$$

5.1.3. Puntuación según la cantidad de seguidores que tiene un usuario

El grado de repercusión de una cuenta de Twitter puede intuirse observando el número de seguidores que esta posee. El hecho de que una cuenta tenga un mayor número de seguidores hace que crezca su credibilidad, es decir, ello significa que gran cantidad de perfiles consideran interesante su contenido y/o apropiado y, además, quieren estar alerta de sus publicaciones. Algunos estudios estiman que el promedio habitual de una cuenta de Twitter ronda los 707 seguidores [45]. Sin embargo, las cuentas pertenecientes a personajes públicos de gran índole

como actores, músicos o políticos entre otros, poseen una cantidad muchísimo más alta que la media, pues presentan una fuerte repercusión mediática. Además, podemos encontrar en contraposición cuentas con pocos seguidores, hecho que puede crear recelo, duda y/o rechazo por parte de muchos usuarios a la hora de ir a interesarse en ese tipo de perfil, pues observan que la cuenta no cuenta con apoyo de suficientes usuarios que respalden su contenido.

Por ello, para este indicador se distinguen como comportamientos sospechosos que la cantidad de seguidores sea menor que la media (menor de 707) y que el número de seguidores sea 0. De esta manera, el indicador partirá de una base de 100 puntos a la cual le serán descontados 50 puntos en el caso de la cantidad de seguidores sea distinta de 0 y menor de 707 y, en el caso de que la cantidad de seguidores sea 0, se descontarán 100 puntos.

$$P(S_u) = 100, \textit{ cuando cantidad\_seguidores} \geq 707$$
$$P(S_u) = 100 - 50 = 50, \textit{ cuando 0 < cantidad\_seguidores} < 707$$
$$P(S_u) = 100 - 100 = 0, \textit{ cuando cantidad\_seguidores} = 0$$

5.1.4. Puntuación según el ratio seguidores/seguidos

Cuando un usuario en Twitter se dispone a seguir el contenido de alguien, un factor que sugestiona mucho este hecho es la cantidad de seguidores que presente dicha cuenta. Normalmente, es frecuente pensar que cuantos más seguidores tenga un usuario en Twitter mayor fiabilidad presenta su contenido, ya que cuenta con una importante y/o considerable cantidad de seguidores que así lo respaldan. Además, aunque la cantidad de seguidores que tenga una cuenta sea relevante, también lo es el número de cuentas a las que el usuario sigue. Sin embargo, estos dos factores cobran mayor sentido estudiándolos conjuntamente, es decir, calculando la relación seguidores/seguidos. Debido al gran número de cuentas falsas o cuentas controladas por bots que existen hoy en día, estudiar esta proporción entre ambos factores es de especial relevancia, ya que en función de cómo sea esta proporción se podrá dar mayor o menos credibilidad a ese usuario. Si una cuenta sigue a un gran número de usuarios lo que se espera de su número de seguidores es una cifra similar o incluso mucho mayor, como ocurre en el caso de las cuentas de personas famosas. Si por el contrario esta cuenta dispone de una cantidad de seguidores ínfima frente a una gran cantidad de cuentas a las que sigue, resulta extraño y, a juicio de los usuarios que visiten su perfil, pone en entredicho su credibilidad como usuario válido y fiable en la plataforma, ya que no cuenta con un respaldo suficiente de seguidores que apoyen su contenido. Por ello, para este indicador se propone estudiar el ratio o proporción existente entre el número de seguidores y el número de seguidos que presente una cuenta. Para ello, se plantean las siguientes proporciones a tener en cuenta [46]:

- Ratio 10 (100 seguidores/10 seguidos): esta proporción sería la considerada ideal en Twitter, es decir, si se tiene un mayor número de seguidores que de gente a la que se sigue, ello significa que existen usuarios que consideran dicho perfil interesante y que, por tanto, quieren continuar estando al tanto de las publicaciones y contenidos de dicha cuenta, independientemente de que no se les devuelva el seguimiento. Este tipo de ratio se puede encontrar frecuentemente en cuentas pertenecientes a personas famosas.
- Ratio 1 (100 seguidores/100 seguidos): esta proporción en muchas ocasiones se da cuando una cuenta devuelve el seguimiento a todos aquellos usuarios que le siguen y/o cuando una cuenta genera un efecto psicológico positivo y provoca que las cuentas a las que sigue le devuelvan el seguimiento [47]. Este valor se considera el más adecuado para aquellas cuentas que busquen crecer su número de seguidores.
- Ratio 0.1 (10 seguidores/100 seguidos): esta proporción es la que se considera más negativa, pues significa que el usuario sigue a muchas cuentas pero en contraposición, pocas de ellas consideran interesante su contenido como para devolverle el seguimiento. Este tipo de cuentas suelen ser propensas a ser revisadas y/o suspendidas por la plataforma de Twitter, pues en muchas ocasiones se trata de cuentas de spam.

Así pues, una vez conocidas las anteriores proporciones, el ratio ideal de seguidores/seguidos para una cuenta sería aquel que se acerque a 1 o incluso supere este valor. De esta manera, cuanto mayor sea este valor de ratio, mayor será por tanto el apoyo que los seguidores brindan al contenido de ese usuario.

Por ello, para este indicador se distingue como perfil sospechoso aquel cuyo ratio sea inferior a 0.75 y como valores positivos se entenderán aquellos que presenten un ratio entre 0.75 y 2, considerando todo ratio que supere al anterior rango como muy positivo. De esta manera, el indicador partirá de una base de 100 puntos (partiendo de la base de un ratio muy positivo) a la cual le serán descontados 50 en el caso de que el ratio sea mayor o igual que 0.75 y menor o igual que 2 y, en el caso de que el ratio sea menor de 0.75 la cantidad a descontar será de 100 puntos.

$$P(R_{ss}) = 100, \textit{ cuando ratio } (\frac{seguidores}{seguidos}) > 2$$

$$P(R_{ss}) = 100 - 50 = 50, \textit{ cuando } 0.75 \leq \textit{ratio } (\frac{seguidores}{seguidos}) \leq 2$$

$$P(R_{ss}) = 100 - 100 = 0, \textit{ cuando ratio } (\frac{seguidores}{seguidos}) < 0.75$$

5.1.5. Puntuación según la antigüedad de la cuenta

En la actualidad no es raro encontrar cuentas falsas creadas con el fin de difundir determinada información acompañando a campañas de marketing, difundir información falsa en la red social o aumentar la cantidad de seguidores de otra cuenta entre otros. Por ello, analizar la antigüedad de una cuenta es un factor relevante, puesto que una cuenta nueva que exista desde hace pocos meses siempre generará una mayor desconfianza que la cuenta de un usuario que muestre una antigüedad de varios años en la plataforma.

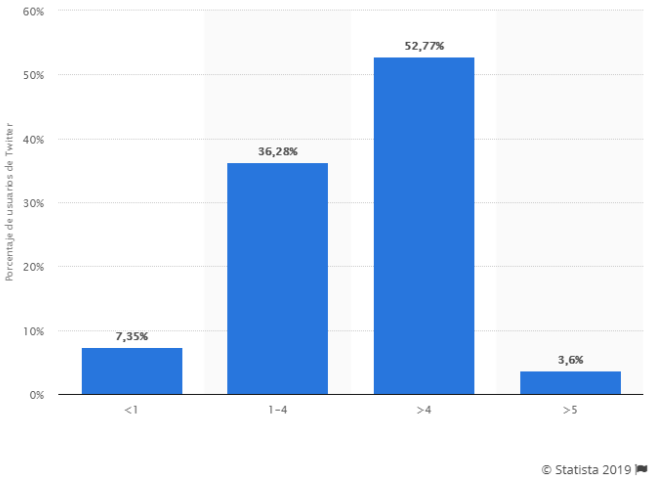


Figura 18. Antigüedad de las cuentas de los usuarios de la red social Twitter en España en 2016. (Fuente https://www.statista.com/)

Analizando estadísticas referentes a los perfiles de las cuentas de Twitter de España en el año 2016 [48], se observa que un poco más de la mitad de las cuentas (52.77%) tenían una antigüedad de más de 4 años. Por otra parte, las cuentas con menos de un año de antigüedad no suponían ni el 10% del conjunto, hecho que denota un escaso crecimiento en el número de usuarios de Twitter de manera anual. Además, es necesario hacer hincapié en que en la actualidad crear una cuenta nueva en la red social no es una tarea tan sencilla como antaño debido a las medidas que recientemente está tomando Twitter para combatir el spam y la manipulación de contenido en la plataforma [49]. Así pues, se destacan principalmente medidas para el estudio de la veracidad de las cuentas nuevas como son la mejora de registro e identificación de usuarios, la búsqueda de signos de registros automatizados en los perfiles o la aplicación de sistemas de detección de comportamientos maliciosos entre otros. Con estas medidas, Twitter se encarga de investigar y localizar este tipo de perfiles falsos con el fin de eliminarlos de la plataforma.

Por ello, para este indicador se distingue como comportamiento sospechoso que la antigüedad de la cuenta sujeta a estudio sea menor de 1 año. De esta manera, el indicador partirá de una

base de 100 puntos (partiendo de que la cuenta presente una antigüedad mayor o igual a 1 año) a la cual le serán descontados 100 puntos en el caso de que no supere dicho año de antigüedad.

$$P(A_c) = 100, \textit{ cuando antigüedad\_cuenta} \geq 1 \textit{ año}$$
$$P(A_c) = 100 - 100 = 0, \textit{ cuando antigüedad\_cuenta} < 1 \textit{ año}$$

5.1.6. Puntuación según si el usuario dispone de una descripción en su perfil o no

Cuando un usuario real se crea una cuenta, normalmente, suele añadir una breve descripción a su perfil. Muchas cuentas falsas o controladas por bots, al ser creadas, únicamente dejan los parámetros por defecto del perfil, incluyendo el campo de la descripción a vacío.

Por ello, para este indicador se distingue como comportamiento sospechoso que la cuenta sujeta a estudio no haya personalizado la descripción de su perfil y haya dicho campo vacío tal y como viene por defecto. De esta manera, el indicador partirá de una base de 100 puntos (partiendo de que la cuenta presente una descripción, siendo la cadena distinta de vacío) a la cual le serán descontados 100 puntos en el caso de que no se haya especificado una descripción para ese perfil.

$$P(D_p) = 100, \textit{ cuando tamaño\_cadena(descripción)} > 0$$
$$P(D_p) = 100 - 100 = 0, \textit{ cuando tamaño\_cadena(descripción)} = 0$$

5.1.7. Puntuación según si el usuario ha compartido su localización geográfica o no

Al igual que en el caso del indicador anterior, un usuario al crearse una cuenta puede dejar por defecto o no sus parámetros de personalización de perfil. Es frecuente que en los casos en los que se trate de una cuenta falsa o controlada por algún tipo de software no se preste atención a rellenar estos campos de personalización. En este caso este indicador se encarga de calcular la puntuación en base a si se ha especificado alguna región geográfica o, de lo contrario, se ha dejado el campo tal y como viene por defecto vacío. Por ello, para este indicador se distingue como comportamiento sospechoso que no se haya especificado una localización. De esta manera, el indicador partirá de una base de 100 puntos (partiendo de que la cuenta presente algún valor para este campo) a la cual le serán descontados 100 puntos en el caso de que la cadena sea vacía.

$$P(L_p) = 100, \textit{ cuando tamaño\_cadena(localización\_geográfica)} > 0$$
$$P(L_p) = 100 - 100 = 0, \textit{ cuando tamaño\_cadena(localización\_geográfica)} = 0$$

5.1.8. Puntuación según si el usuario ha personalizado el diseño de su perfil o no

Este indicador resulta ser, al igual que los dos indicadores anteriores, un parámetro de personalización de la cuenta de un usuario. En este caso, cuando se crea una cuenta de 0, ésta viene por defecto sin ningún tipo de personalización de colores en su perfil. Normalmente, las cuentas falsas o aquellas controladas por bots, no suelen modificar el decorado del perfil y lo dejan por defecto. Por ello, para este indicador se distingue como comportamiento sospechoso que no se hayan personalizado los colores del perfil. De esta manera, el indicador partirá de una base de 100 puntos (partiendo de que se han customizado los colores de la cuenta) a la cual le serán descontados 100 puntos en el caso de que lleve el diseño por defecto.

$$P(P_p) = 100, \textit{ cuando personalización\_por\_defecto = FALSE}$$
$$P(P_p) = 100 - 100 = 0, \textit{ cuando personalización\_por\_defecto = TRUE}$$

5.1.9. Puntuación según la cantidad de "likes" dados por el usuario

Normalmente un usuario activo realiza tanto publicaciones propias, como da retweets o da me gusta de forma habitual. Dar favorito a una publicación significa que al usuario le ha gustado dicho contenido, pero por alguna razón no quiere compartir en su perfil dicha publicación y colaborar en su difusión. Sin embargo, Twitter ha realizado varios cambios sobre esta utilidad de dar like en la plataforma (anteriormente conocidos como favoritos) y mucha gente no les presta la misma atención. Por ello, normalmente una cuenta perteneciente a un usuario real tendrá una cantidad de me gustas dados más o menos grande, pero tendrá. Normalmente las cuentas controladas por bots o creadas para determinados fines como la difusión mediática de un determinado contenido, se centran en eso, en difundir un determinado contenido, dejando a un lado la tarea de dar me gustas.

Por ello, para este indicador se distingue como comportamientos sospechosos que un usuario no haya dado ningún me gusta (cantidad de likes con valor a 0) y que en el caso de que tenga likes, este número sea inferior a 200. De esta manera, el indicador partirá de una base de 100 puntos (partiendo de que la cuenta ha dado como mínimo 200 me gustas) a la cual le serán descontados 50 puntos en el caso de que la cantidad de likes sea distinta de 0 y menor a 200 y, finalmente, en el caso de que la cuenta no haya dado ningún like se le restarán 100 puntos.

$$P(N_l) = 100 - 100 = 0, \textit{ cuando } numero\_likes = 0$$
$$P(N_l) = 100 - 50 = 50, \textit{ cuando } 0 < numero\_likes < 200$$
$$P(N_l) = 100, \textit{ cuando } numero\_likes \geq 200$$

5.1.10. Puntuación según la media de publicaciones en base a la antigüedad de la cuenta en días

En dos indicadores anteriores se habla de la cantidad de retweets realizados por una cuenta y de la cantidad de publicaciones propias de manera diaria. En este indicador se pretende calcular la media de publicaciones diarias (retweets y tweets propios) que ha realizado y/o compartido un usuario desde que se creó dicha cuenta (tomando 365 días por año natural). De esta manera, la media diaria será calculada siguiendo la siguiente fórmula:

$$Mpa = \frac{total\_publicaciones}{365 \cdot antigüedad\_cuenta}$$

Donde:

$M_{pa} -$ Media de publicaciones diarias efectuadas según antigüedad de la cuenta.

Por ello, para este indicador se distingue como comportamiento sospechoso que el desglose diario de publicaciones en base a la antigüedad de la cuenta se traduzca en un número muy bajo de interacciones o bien totalmente lo contrario, que el número medio de interacciones sea muy significativo. Así pues, se considerará negativamente si el usuario presenta una media por debajo de 15 interacciones diarias o bien si por el contrario presenta una cantidad muy elevada (más de 30). Para ello se base de una puntuación inicial de 500 tomando como base que se presenta una media de interacciones correcta y, en el caso de que se superen las 30 publicaciones diarias o bien sea un número comprendido entre 5 y 14 la puntuación bajará a los 100 puntos. En los casos en los que se supere la cantidad de 40 interacciones diarias o bien la cantidad de publicaciones sea inferior a 5, la puntuación será 0.

$$P(T_a) = 500 - 500 = 0, \textit{ cuando } M_{pa} < 5$$
$$P(T_a) = 500 - 400 = 100, \textit{ cuando } 5 \leq M_{pa} < 15$$
$$P(T_a) = 500, \textit{ cuando } 15 \leq M_{pa} \leq 30$$
$$P(T_a) = 500 - 400 = 100, \textit{ cuando } 30 < M_{pa} \leq 40$$
$$P(T_a) = 500 - 500 = 0, \textit{ cuando } M_{pa} > 40$$

5.1.11. Puntuación según el resultado obtenido para el usuario en Botometer

Hoy en dia descartar que una cuenta esté manejada por algún tipo de software no es una tarea sencilla de llevar a cabo. Por ello, para este indicador se pretende calcular orientativamente este hecho utilizando para ello el algoritmo Botometer [29]. Así pues, las puntuaciones obtenidas para este cálculo oscilarán entre el 0 y el 5, donde cuanto más cercana sea la puntuación a 0 mayor será la probabilidad de que se trate de una cuenta manejada por una persona real y, por

el contrario, cuanto mayor sea la puntuación mayor probabilidad habrá de que la cuenta sujeta a estudio esté siendo manejada por un bot. Por ello, para este indicador se distingue como comportamiento sospechoso que la puntuación obtenida en Botometer sea mayor de 1 punto. De esta manera, el indicador partirá de una base de 500 puntos (partiendo de que se trate de una cuenta controlada por una persona real) a la cual le serán descontados puntos el caso de que se obtenga una puntuación mayor que 1, restando 500 puntos del total.

$$P(P_b) = 500, \textit{ cuando } nota\_botometer \leq 1$$
$$P(P_b) = 500 - 500 = 0, \textit{ cuando } 1 < nota\_botometer \leq 5$$

5.2. Factores determinantes

Para obtener las **puntuaciones de los factores determinantes** se parte de una puntuación base de 100 puntos para cada indicador.

En el caso de que el aspecto que está siendo evaluado no contribuya positivamente con la credibilidad del usuario, pasará a tener como valor 1, es decir, no provocará ninguna alteración en la puntuación final, pero si lo hará de forma notoria en su defecto.

5.2.1. Puntuación según si la cuenta está verificada o no

El hecho de que una cuenta de interés público esté verificada confirma la autenticidad de esta, siendo normalmente perfiles pertenecientes al ámbito musical, cinematográfico, empresarial, deportivo o periodístico entre otros [38]. Este factor resulta determinante, puesto que conseguir la verificación de Twitter demuestra que dicha cuenta pertenece a la persona u organización a la que hace referencia, es decir, verificando las cuentas Twitter confirma la identidad de estas [39]. De la misma manera, Twitter se reserva el derecho a retirar la verificación en el caso de que se observen intentos de engaño, incitación a la violencia, al odio, al acoso o incumplir alguna regla de Twitter [40]. Así pues, por ello se considera que este indicador resulta ser un factor determinante, puesto que es la propia red social la que ha autorizado y confirmado la identidad legítima de la cuenta. De esta forma, el hecho de que una cuenta esté verificada supondrá un incremento importante sobre la puntuación total de la credibilidad del usuario (se multiplica el total por los 1000 puntos base asignados al indicador) y, en el caso de que no sea así, la puntuación no se verá alterada (se multiplica el total por la unidad).

$$P(V_c) = 1000, \textit{ cuando se trate de una cuenta verificada.}$$

$$P(V_c) = 1, \textit{ cuando se trate de una cuenta no verificada.}$$

5.2.2. Puntuación según si la cuenta tiene foto de perfil o no

Twitter, al igual que muchas otras redes sociales ha sido y sigue siendo víctima de cuentas falsas bien con fines de acoso, spam o cuentas controladas por software entre otros (bots). Este tipo de cuentas con frecuencia comparten como característica no poseer una foto de perfil personalizada, es decir, poseen la imagen por defecto que brida Twitter cuando se crea una cuenta desde cero. Un claro ejemplo que demuestra el rechazo de Twitter a este tipo de cuentas resultan ser las medidas para combatir el acoso anunciadas por la plataforma, entre ellas que los usuarios puedan filtrar todo aquel contenido publicado por cuentas que no hayan personalizado su foto de perfil [41]. Así pues, por ello se considera que este indicador resulta ser un factor determinante, puesto que es la propia red social la que está tomando medidas para evitar todo este tipo de cuentas fraudulentas. De esta forma, el hecho de que una cuenta no haya personalizado su foto de perfil supondrá un decremento importante sobre la puntuación total de la credibilidad del usuario (se multiplica el total acumulado por los $\frac{1}{1000}$ puntos base asignados al indicador) y, en el caso de que no sea así, la puntuación no se verá alterada (se multiplica el total por la unidad), puesto que disponer de foto personalizada no garantiza una mayor fiabilidad pero la ausencia de esta sí afecta de forma notoria a la puntuación final.

$P(I_p) = \frac{1}{1000}$, cuando se trate de una cuenta con foto de perfil por defecto.

$P(I_p) = 1$, cuando se trate de una cuenta con foto de perfil personalizada.

-

5.3. Validación de la propuesta

En este apartado realizaremos la validación de nuestro algoritmo, para ello utilizaremos un grupo de datos de control y comprobaremos que los resultados que genera el algoritmo concuerdan con los esperados.

5.3.1. Selección de fuentes

Para la validar el algoritmo propuesto será necesario escoger un conjunto de cuentas que pruebe distintos tipos de perfiles a clasificar. Para ello, se propone la siguiente clasificación para los perfiles de testeo (estas serán cuentas que conocemos perfectamente y sabemos en principio su clasificación, y esperamos que una vez procesadas por nuestro algoritmo, genere un resultado concreto esperado):

Cuentas buenas

- Perfiles reales verificados: se trata de cuentas que pertenecen a personajes o empresas con cierta relevancia en la sociedad (grupos de música, empresas de videojuegos o youtubers entre otros).
- Perfiles reales no verificados: resultan ser cuentas que pertenecen a personas u organizaciones reales pero que no presentan una relevancia tan importante como las anteriores en la red social y no están verificadas.

Cuentas malas

- Perfiles falsos o controlados por bots: serán cuentas que presenten claros síntomas de inactividad, comportamientos de spam o inciten a pensar que están siendo controlados por algún tipo de software.

Así pues, para probar cada una de estas clasificaciones, se seleccionará un conjunto de cuentas en las que se observen cualidades que casen con cada tipo de perfil, con el fin de verificar el correcto funcionamiento del algoritmo propuesto a la hora de clasificar los distintos usuarios.

5.3.2. Búsqueda y selección de cuentas de prueba

Para poder probar clasificación de perfiles propuesta en este trabajo, se procede a escoger cuentas a mano que presenten características claras de que pertenecen a uno u otro grupo de los perfiles planteados. Así pues, las cuentas escogidas para cada tipo de perfil de usuario son las siguientes:

Perfiles reales o empresas reales

Perfiles reales verificados que pertenecen a personajes o empresas reconocidos

- NintendoES: cuenta oficial de Twitter de la empresa de videojuegos Nintendo en España.
- Slipknot: cuenta oficial del grupo musical Slipknot.
- DeadByBHVR: cuenta oficial sobre el videojuego Dead By Daylight que gestiona la empresa desarrolladora Behaviour.
- Menos_trece: creador de contenido en plataformas como Youtube o Twitch.

Perfiles reales no verificados

- Cahlaflour: creadora de contenido en la plataforma Twitch no muy conocida que presenta actividad en la red social de manera diaria.
- MelonieMac: creadora de contenido en las plataformas Facebook, Twitch y Youtube que presenta actividad en la red social de manera diaria y no es muy conocida.

Perfiles que se sospecha que son falsos o controlados por bots

- Merm114: cuenta propia creada de prueba, que no presenta foto de perfil, ni tampoco seguidores y apenas tiene interacciones o publicaciones.
- Un_bot_kawaii: cuenta que reconoce ser un bot en su descripción e indica quién es su autor.
- Botdelcuerpo: cuenta que reconoce ser un bot en su descripción e indica quién es su autor.
- cuentafalsa385: cuenta que, como su propio nombre indica, se intuye que es falsa, ya que además no muestra foto de perfil y apenas presenta actividad alguna, así como seguidores.

Freecouponarena: cuenta que se intuye que está siendo controlada por algún tipo de software ya que presenta una cantidad de publicaciones de manera diaria importante, presentando todas ellas siempre algún enlace a un sitio web y no muestran ni retweets ni favoritos.

5.3.3. Procesamiento de datos

Para validar la propuesta detallada anteriormente, será necesario utilizar una pequeña parte del código que más abajo en el siguiente punto se explicará en detalle. Esta parte necesaria a desarrollar, resultar ser el código que permitirá acceder a la API de Twitter para recoger los datos de los usuarios sobre las 23:00 horas cada día, con los que se calculará la puntuación final $P(CC_t)$. Dado que esta porción de código es muy simple, y hace uso de librerías de terceros que ya están validadas, no entraremos en las pruebas y validación de esta porción de código. Este código a grandes rasgos lo que hace es recoger los datos de cada usuario del conjunto detallado en el punto anterior, a esa hora tan tardía de cada día, puesto que hay factores que dependen de contadores acumulativos durante el día como el número de publicaciones o el número de retweets que ha generado ese usuario para ese día. Como se ha dicho, en el siguiente apartado de implementación se mostrará el detalle del código fuente.

fecha_hora_medicion	cuenta	id	nombre	seguidores	amigos	verificada	fecha_creacion	descripcion	localizacion	imagen_por_defecto	total_favoritos_cuenta	tota
19/07/2019 23:00:02	botdelcuerpo	9,29808E+17	bot del cuerpo	728	46	FALSE	12/11/2017 20:27:40	Hola soy el bot del cuerpo. Soy un proyecto de @andre...		FALSE	50	
19/07/2019 23:00:02	Cahlaflour	46033797	Cahlaflour	12581	958	FALSE	10/06/2009 4:10:07	Cahleesi | @Twitch Partner | Streamer of the Year Fina...		FALSE	40952	
19/07/2019 23:00:02	cuentafalsa385	2457402595	Cuenta Falsa	4	142	FALSE	22/04/2014 1:03:29			TRUE	0	
19/07/2019 23:00:02	DeadByBHVR	4850837842	Dead by Daylight	264977	353	TRUE	26/01/2016 19:08:12	#DeadbyDaylight is an asymmetrical multiplayer horror...	Montréal, Québec	FALSE	897	
19/07/2019 23:00:02	freecouponarena	2943099135	Free Coupon Arena	686	670	FALSE	27/12/2014 2:21:29	https://t.co/QfOaxsquCH: Find free coupons and promo...	Estados Unidos	FALSE	3	
19/07/2019 23:00:02	MelonieMac	42361118	Melonie Mac🐝❤	102838	1769	FALSE	25/05/2009 6:07:45	Host for @GameStop TV. @facebookgaming partnered...	Dallas, Texas	FALSE	59822	
19/07/2019 23:00:02	menos_trece	2784032170	Menos Trece	112913	203	TRUE	25/09/2014 18:03:08	Subo vídeos a Youtube y hago directos en Twitch, teng...	Barcelona	FALSE	278	
19/07/2019 23:00:02	merm114	1,08237E+18	merm11	0	2	FALSE	07/01/2019 20:24:28			TRUE	11	
19/07/2019 23:00:02	NintendoES	494308508	Nintendo España	437307	6	TRUE	16/02/2012 19:17:00	Bienvenido al Twitter oficial de Nintendo España con in...		FALSE	10	
19/07/2019 23:00:02	Slipknot	108361327	Slipknot	2093476	11	TRUE	25/01/2010 18:09:25	WE ARE NOT YOUR KIND: New Album August 9. Pre-ord...	Des Moines, IA	FALSE	264	
19/07/2019 23:00:02	un_bot_kawaii	1,05338E+18	(/◕ヮ◕)/*:･ﾟ✧ un bot kawaii	629	8	FALSE	19/10/2018 20:36:12	un bot kawaiiparte de la familia de bots de @brokenen...	japon	FALSE	3	

Figura 19. Captura del fichero con algunos de los datos de los usuarios recopilados de la API (Fuente propia)

Así pues, para esta validación inicial de la propuesta se programa la recopilación de datos de las cuentas de usuario que se han escogido para ello en el apartado anterior durante el periodo de un mes (30 días). De esta manera, puesto que se trata de un total de 11 cuentas a estudiar, si multiplicamos esta cantidad por el número de días que se van a extraer datos, se deberá obtener un total de 330 registros de datos. Se propone un periodo de recopilación de datos de tantos días ya que, cuanto más tiempo transcurra, mayor fiabilidad tendrá la tendencia resultado de cada una de las cuentas.

Así pues, una vez recogidos los datos de las distintas cuentas y comprobado que hemos obtenido el número de registros final correctamente, se procede a calcular la puntuación final de forma manual para cada una de las mismas a mano. De esta forma las puntuaciones $P(CC_t)$ para cada cuenta con su respectiva tendencia resultarían de la siguiente manera:

Cuentas buenas

➔ Perfiles reales verificados

NintendoES

Si analizamos la gráfica obtenida para esta cuenta, observamos una tendencia lineal más o menos constante pero que presenta ciertos picos de actividad momentáneos en los que su $P(CC_t)$ se dispara casi al doble. Puesto que se trata de una empresa que se dedica al desarrollo de consolas y videojuegos, probablemente dichos días que presentan esos picos de puntuación tan importantes sean debido a algún lanzamiento o campaña promocional de alguno de sus productos que se lanzara sobre esas fechas, pudiendo aumentar considerablemente por ello la actividad de esta cuenta en esos momentos puntuales.

Además, puesto que se trata de una cuenta verificada, podemos observar que sus puntuaciones son extremadamente altas, ya que el factor determinante que puntúa este hecho tiene un peso importante en el cálculo de su $P(CC_t)$ final. *Así pues, a esta cuenta se le asigna la categoría de "cuenta buena".*

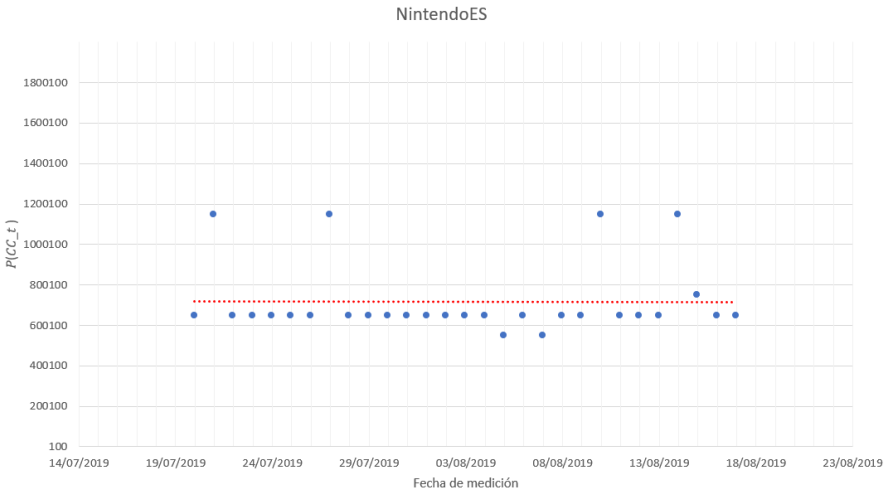


Figura 20. Tendencia del $P(CC_t)$ durante 1 mes para la cuenta "NintendoES" (Fuente propia)

Slipknot

Si observamos la gráfica obtenida para esta cuenta, observamos una tendencia lineal más o menos constante, sin embargo, presenta bastantes picos en su $\mathrm{P(CC_t)}$ a lo largo del transcurso del mes. Se observa, además, un par de bajadas puntuales de su puntuación total casi a mitad del mes de agosto. Este hecho puede ser debido a que en esas fechas el grupo musical ya había terminado su gira de conciertos y, por tanto, entra dentro de lo previsible el hecho de que su actividad pueda menguar en esas fechas por este hecho.

Por otra parte, al igual que en el caso de la cuenta anterior, su $\mathrm{P(CC_t)}$ resulta muy alto puesto que el factor determinante asociado a que una cuenta esté verificada aumenta enormemente la puntuación final. Así pues, a esta cuenta se le asigna la categoría de "cuenta buena".

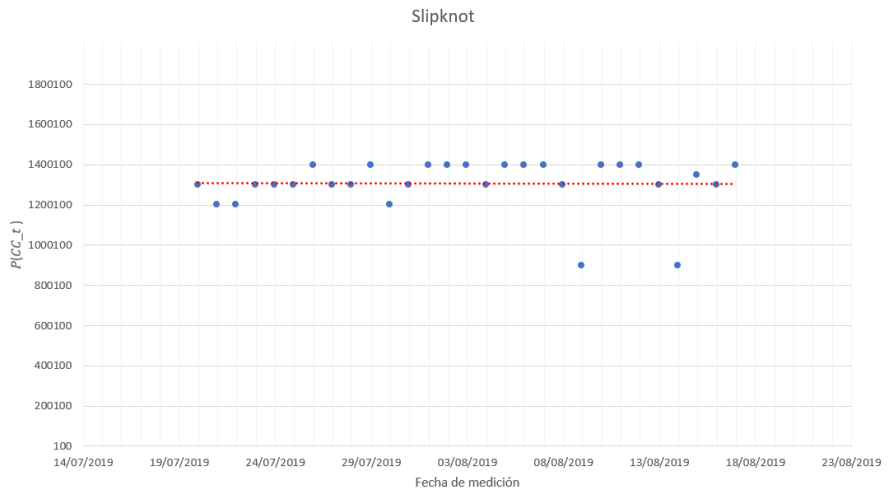


Figura 21. Tendencia del $P(CC_t)$ durante 1 mes para la cuenta "Slipknot" (Fuente propia)

DeadByBHVR

Si observamos la gráfica obtenida para esta cuenta, observamos una tendencia lineal más o menos constante y que, a diferencia de las dos cuentas anteriores, prácticamente todo el mes que se monitorizó apenas tuvo variaciones en su $\mathrm{P}(\mathrm{CC_t})$. El momento puntual en el que se sufrió una subida importante en su puntuación final fue un día durante la primera mitad de agosto, con una subida de casi 600.000 puntos con respecto a su línea de tendencia, poniéndose con un $\mathrm{P}(\mathrm{CC_t})$ muy cercano a los 1.400.000 puntos. Puesto que se trata de la cuenta oficial del juego Dead By Daylight, probablemente ese día se anunciase un parche o expansión nueva para el juego, hecho que aumentó considerablemente la actividad de la cuenta para esa fecha. Así pues, a esta cuenta se le asigna la categoría de "cuenta buena".

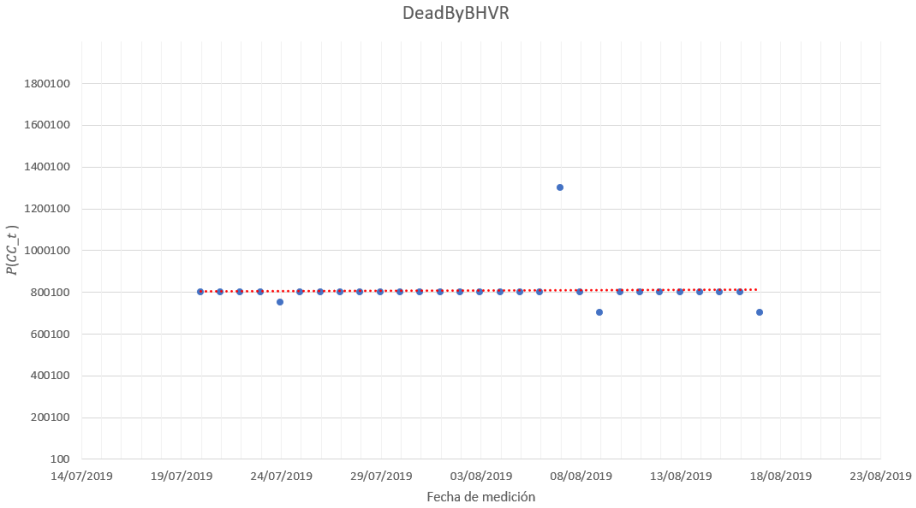


Figura 22. Tendencia del $P(CC_t)$ durante 1 mes para la cuenta "DeadByBHVR" (Fuente propia)

Menos_trece

Si observamos la gráfica obtenida para esta cuenta, observamos una tendencia lineal más o menos constante y que, a diferencia del resto de cuentas anteriores, es la única que no pretenda picos importantes en su $\mathrm{P}(\mathrm{CC_t})$, respetando notoriamente durante todo el mes que se monitorizó la cuenta el valor de su puntuación final. Las variaciones más importantes que presenta se acercan a los 100.000 puntos más o menos respecto a su línea de tendencia, un margen de error poco significativo si tenemos en cuenta las variaciones tan notorias que sufrían algunas de las cuentas anteriores. Así pues, a esta cuenta se le asigna la categoría de "cuenta buena".

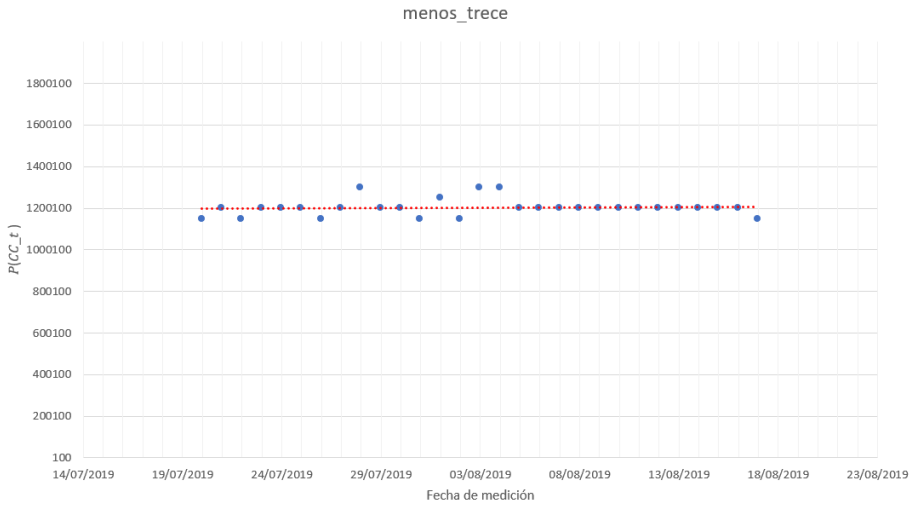


Figura 23. Tendencia del $P(CC_t)$ durante 1 mes para la cuenta "menos_trece" (Fuente propia)

Como conclusión sobre el $\mathrm{P(CC_t)}$ de estos perfiles reales verificados, podemos afirmar que este tipo de cuentas suelen superar los 500.000 puntos en su puntuación final, ya que el hecho de ser cuentas verificadas favorece enormemente este incremento de puntos sobre el total y, además, muchas veces presentan una actividad muy constante e incluso como se ha podido observar, pueden presentar picos muy abruptos de subidas de puntuación derivadas o impulsadas por algún anuncio o campaña promocional, ya que se tratan de cuentas pertenecientes a personajes o empresas reconocidas.

➔ Perfiles reales no verificados

<u>Cahlaflour</u>

Si observamos la gráfica obtenida para esta cuenta, se observa un decrecimiento en su tendencia del $\mathrm{P(CC_t)}$ y que empieza a acentuarse más a partir del mes de agosto. Este hecho podría deberse a que en esa época del año mucha gente suele aprovechar para desplazarse y/o ausentarse por motivos vacacionales.

Por otra parte, presenta picos eventuales con bajadas en su puntación final pero que son apenas perceptibles puesto que son diferencias con un margen de error pequeño de tan solo 100 puntos. Así pues, a esta cuenta se le asigna la categoría de "cuenta buena".

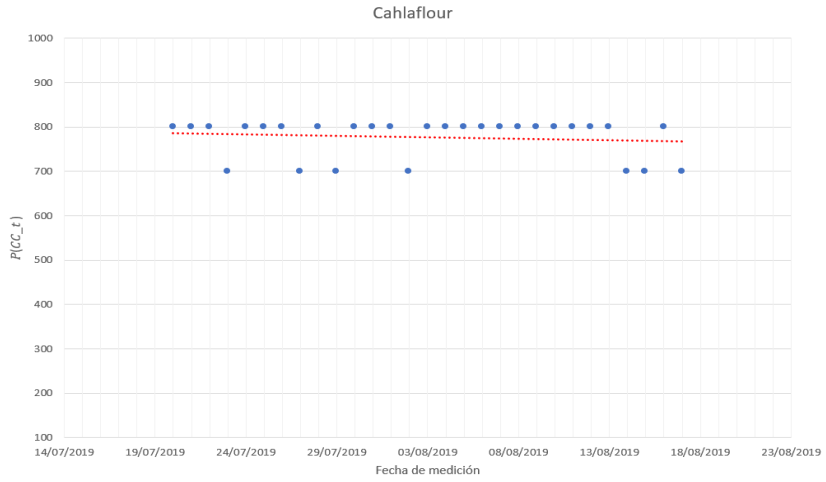


Figura 24. Tendencia del $P(CC_t)$ durante 1 mes para la cuenta "Cahlaflour" (Fuente propia)

MelonieMac

Si observamos la gráfica obtenida para esta cuenta, al igual que en el caso de la cuenta anterior, presenta un decrecimiento en su tendencia a partir del mes de agosto, suponiendo al igual que en el caso anterior, que se deriva de motivos relacionados con las vacaciones o el descanso al encontrarse en esas fechas.

Por otro lado, esta cuenta presenta una actividad más cambiante y variante que la anterior, ya que es cierto que no presenta variaciones significativas en su $\mathrm{P(CC_t)}$ pero si que presenta subidas y bajadas más frecuentes que en el caso anterior, aunque con poco margen de error: en torno a los 100 puntos como el usuario anterior. Así pues, a esta cuenta se le asigna la categoría de "cuenta buena".

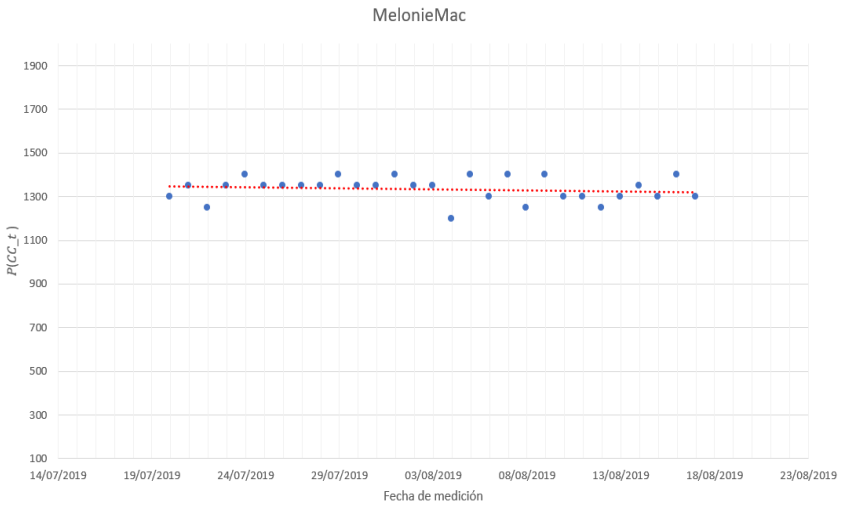


Figura 25. Tendencia del $P(CC_t)$ durante 1 mes para la cuenta "MelonieMac" (Fuente propia)

Como conclusión sobre el $P(CC_t)$ de estos perfiles reales no verificados, podemos afirmar que este tipo de cuentas suelen superar los 600-700 puntos en su puntuación final y, además, suelen presentar variaciones en este valor, bien de manera puntual o bien de una manera más frecuente, ya que al tratarse de personas reales la actividad puede variar según periodos del año, vida personal u otros aspectos, nunca será una línea perfecta de puntos consecutivos debido a esta aleatoriedad humana.

Cuentas malas

➔ Perfiles que se sospecha que son falsos o controlados por bots

Botdelcuerpo

Si observamos la gráfica obtenida para esta cuenta, se puede apreciar que la tendencia de su puntuación final en el tiempo presenta un ligero decrecimiento paulatino, según las puntuaciones obtenidas para su $\mathrm{P(CC_t)}$ durante el mes de monitorización.

Además, destaca el hecho de que sus puntuaciones en la gráfica formen una línea consecutiva de puntos formando una línea horizontal, hecho que indica que presenta un comportamiento y unas características que no varían apenas con el tiempo, se mantienen muy estáticas. Un comportamiento tan lineal en el tiempo y que no presenta ningún tipo de variación puntual, inclina a pensar que no se trata de una cuenta controlada por una persona, es decir, se trata de una cuenta controlada por algún tipo de software. Así pues, a esta cuenta se le asigna la categoría de "cuenta mala".

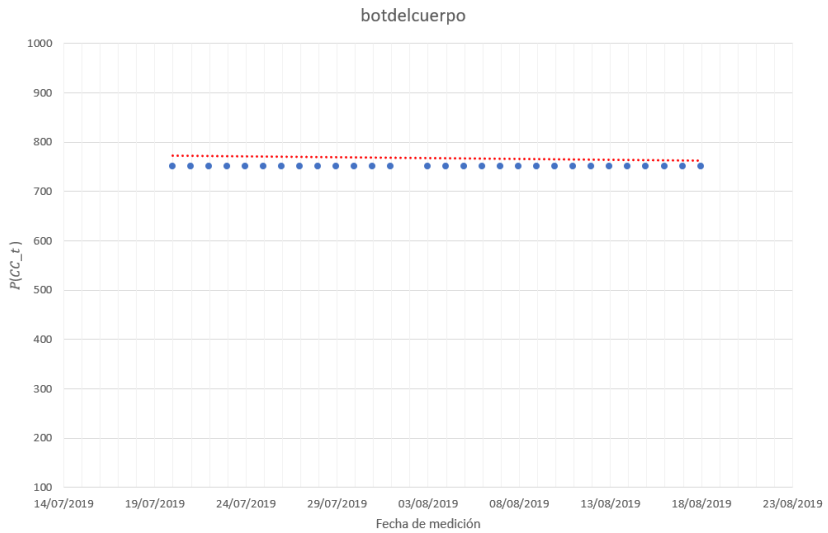


Figura 26. Tendencia del $P(CC_t)$ durante 1 mes para la cuenta "botdelcuerpo" (Fuente propia)

Cuentafalsa385

Si observamos la gráfica obtenida para esta cuenta, se puede apreciar que la tendencia de su puntuación final en el tiempo apenas varía al igual que en el caso de la cuenta anterior. Presenta una puntuación muy lineal y que apenas tiene variaciones, formando una línea horizontal de puntos. Nuevamente observamos que ante un $\mathrm{P(CC_t)}$ tan estable en el tiempo, descarta enormemente de que se trate de una cuenta real y controlada por una persona o entidad verídicas.

Otro hecho que llama enormemente la atención es la puntuación tan excesivamente baja que presenta su $\mathrm{P(CC_t)}$. Hay factores no determinantes que valoran características muy básicas que se encuentran presentes en los perfiles reales de manera habitual (personalización de los colores, imagen de perfil personalizada, etc.) y el hecho de que esta cuenta haya presentado una puntuación tan baja indica que apenas ha habido factores para los que el algoritmo diseñado le haya asignado puntos (no cumple los criterios marcados). Así pues, a esta cuenta se le asigna la categoría de "cuenta mala".

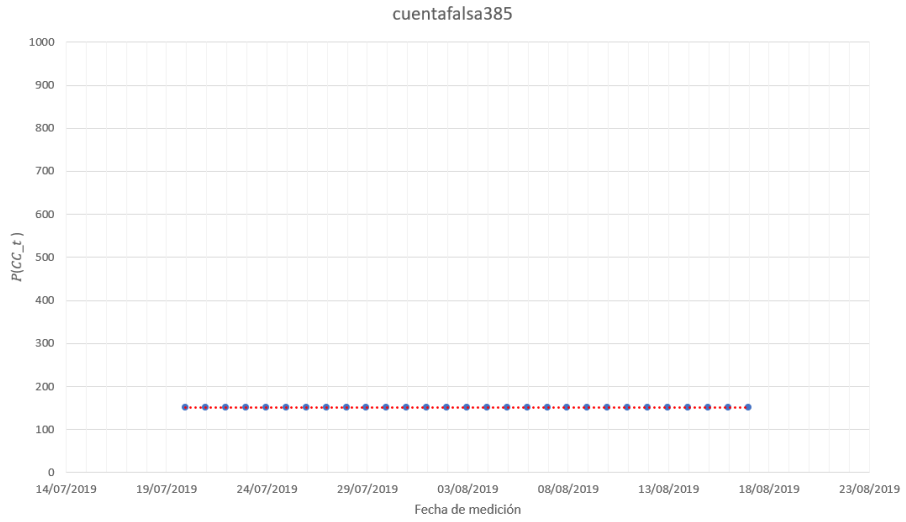


Figura 27. Tendencia del $P(CC_t)$ durante 1 mes para la cuenta "cuentafalsa385" (Fuente propia)

Freecouponarena

Si observamos la gráfica obtenida para esta cuenta, se puede apreciar que la tendencia de su puntuación final decrece con el transcurso de los días. Su $\mathrm{P(CC_t)}$ presenta pequeñas y ligeras variaciones a lo largo del tiempo, no superando en ningún caso un error mayor de 100 puntos con respecto a su tendencia. Esta irregularidad leve en su puntuación final aleja en cierta medida de pensar que se trata de una cuenta controlada por un software.

Sin embargo, su puntuación no alcanza ni los 600 puntos, hecho que indica que cumple a duras penas las condiciones para que el algoritmo le asigne puntos para algunos de los factores determinantes o no determinantes.

En el caso de esta cuenta, que conocemos que por sus contenidos que se trata de una cuenta de spam, sin embargo, podemos observar que en la forma de su comportamiento deberíamos considerarla como "buena", ya que simula el comportamiento de una cuenta real de las que nosotros consideramos como "cuentas buenas", denotando cierta credibilidad y calidad. Por tanto, es necesario tener en cuenta que según el algoritmo propuesto, si una cuenta de Twitter es capaz de simular ser un usuario bueno, generará lo que se conoce como **falso positivo**.

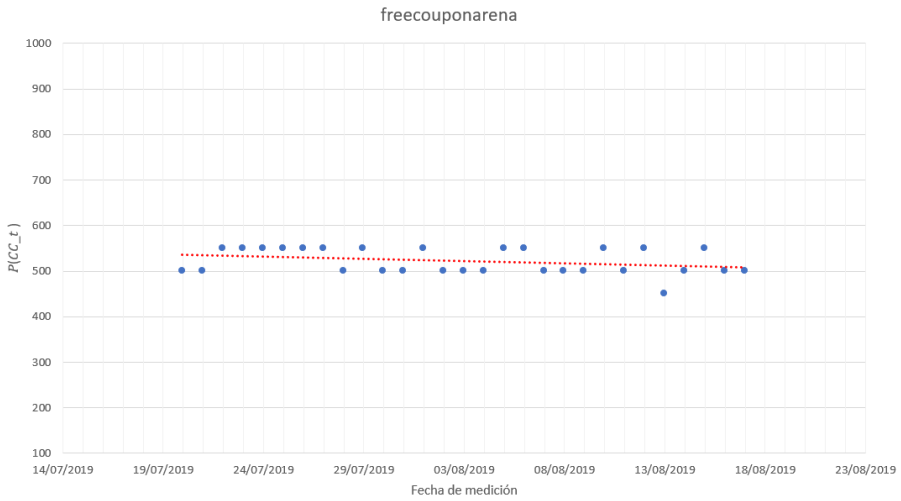


Figura 28. Tendencia del $P(CC_t)$ durante 1 mes para la cuenta "freecouponarena" (Fuente propia)

Merm114

Si observamos la gráfica obtenida para esta cuenta, se puede apreciar que la tendencia de su puntuación final decrece paulatinamente en el tiempo. Inicialmente presenta cierta actividad pero pronto se corta y empieza a decrecer. El resto de los días se va consolidando su puntuación final en una línea consecutiva de puntos de manera horizontal y con $\mathrm{P(CC_t)}$ extremadamente bajo, no llegando siquiera a los 100 puntos. Estas circunstancias pueden ser un claro indicio de que, probablemente, se trate de una cuenta creada en un momento determinado y que poco después fue abandonada, hecho que explicaría la puntuación tan extremadamente baja que presenta. Así pues, a esta cuenta se le asigna la categoría de "cuenta mala".

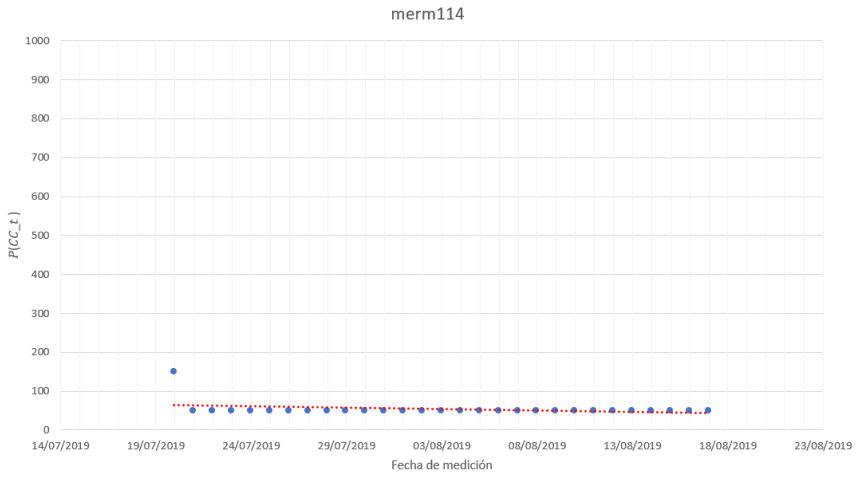


Figura 29. Tendencia del $P(CC_t)$ durante 1 mes para la cuenta "merm114" (Fuente propia)

Un_bot_kawaii

Si observamos la gráfica obtenida para esta cuenta, se puede apreciar que la tendencia de su puntuación final decrece paulatinamente en el tiempo. Además, presenta casi al principio un $P(CC_t)$ atípico durante el tercer día de monitorización. Sin embargo, el resto de días resultan ser muy similares entre sí, casi idénticos y aunque presenten cierta puntuación final ya considerable, podemos observar cómo se forma una línea horizontal consecutiva de puntos. Con ello, al igual que en casos anteriores, podemos deducir que presenta un comportamiento bastante lineal en el tiempo, hecho que lo aleja enormemente de asemejarse al comportamiento humano y lo cual da a pensar que se trata de una cuenta controlada por un software. Por todo ello, a esta cuenta se le otorga la categoría de "cuenta mala".

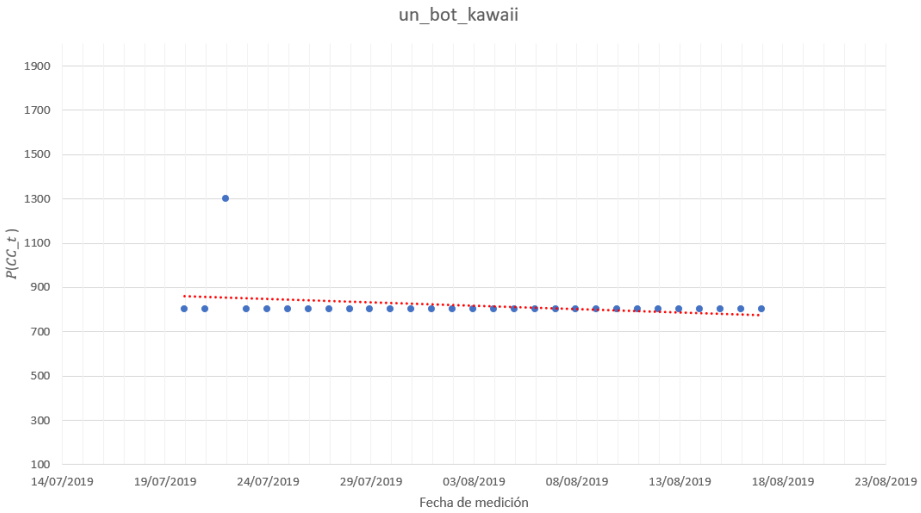


Figura 30. Tendencia del $P(CC_t)$ durante 1 mes para la cuenta "un_bot_kawaii" (Fuente propia)

Como conclusión sobre el $P(CC_t)$ de estos perfiles que se sospecha que son falsos o controlados por bots, podemos afirmar que por norma general suelen presentar una puntuación bastante baja, no llegando apenas a sobrepasar los 800 puntos. Sin embargo, un indicio muy claro que presentan prácticamente todas en común resulta ser la extraña línea de puntos horizontales observable en la gráfica de sus $P(CC_t)$ a lo largo del mes, es decir, presentan un comportamiento prácticamente estático y exacto todos los días, cuya tendencia suele decrecer levemente. Sin embargo, ante situaciones donde aparezcan cuentas con una puntuación extremadamente baja (no sobrepasen los 100 puntos), probablemente estemos lidiando con una cuenta abandonada.

6. Implementación y validación

En este apartado se procede a detallar la implementación paso a paso del algoritmo presentado anteriormente en el apartado de propuesta y, posteriormente se verificará que el código desarrollado es correcto comparando con los resultados previamente obtenidos de forma manual para los distintos $P(CC_t)$ de las cuentas. Además, el hecho de documentar en detalle todas las fases del desarrollo, posibilitaría que cualquier usuario interesado pudiera replicar los experimentos que se han llevado a cabo en este trabajo.

6.1. Preparación del entorno

6.1.1. Creación de máquina virtual con Ubuntu

Primeramente, se procede a crear una máquina virtual donde se llevará a cabo la realización del proyecto. Con ello se pretende generar un entorno de desarrollo y pruebas totalmente limpio, exento de posibles interferencias con otros software, librería o cualquier tipo de dependencia, asegurándonos así que no se producirán problemas más allá de los relacionados con el desarrollo.

En este caso se ha escogido montar una máquina virtual con el sistema operativo Ubuntu, el cual resulta ser una distribución de Linux de código abierto. Se ha optado por este sistema operativo ya que, tal y como se indica en la documentación oficial para desarrolladores de su página web [50], resulta muy ligero para ejecutarlo en una máquina virtual o de manera nativa. Además, posee un amplio catálogo de bibliotecas de desarrollo, hecho que facilita enormemente cualquier tarea a los desarrolladores. En este caso se ha configurado Ubuntu en una de sus versiones más estables, la “16.04 LTS” [51].

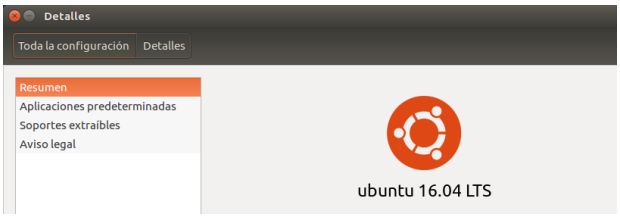


Figura 31. Captura con la versión de Ubuntu instalada en la máquina (Fuente propia)

6.1.2. Python 3.5 y Pycharm 2018.3

Una vez configurada la máquina virtual con el sistema operativo correspondiente, será necesario preparar el entorno para el desarrollo del código del algoritmo propuesto. Así pues, para el desarrollo de este proyecto se opta por el lenguaje de programación Python en su versión 3.5. Para ello se procederá a instalar Python vía terminal, usando para ello los siguientes comandos [52]:

```
$ sudo apt-get update
```

```
$ sudo apt-get install Python 3.5
```

Una vez hecho esto, para comprobar que todo se ha instalado correctamente se procederá a revisar la versión de Python que se tiene instalada en la máquina mediante el siguiente comando:

```
$ python3 -V
```

Si todo ha salido correctamente, tras ejecutar dicho comando se deberá obtener como salida por pantalla la versión de Python recién instalada, en este caso Python 3.5.

Tras ello, se procederá a instalar un entorno de desarrollo integrado (IDE) para desarrollar proyectos en lenguaje Python. Para ello se escoge como IDE el programa PyCharm, actualmente conocido como uno de los mejores y más completos para programar en código Python [54].

Así pues, para obtener este programa se procede a su descarga a través de su sitio web oficial [53]. En este caso se procederá a descargar e instalar PyCharm en su versión "Community 2018.3", la cual resulta gratuita y ofrece los elementos necesarios y básicos para desarrollar en código Python.

Figura 32. Captura con la versión de Pycharm instalada en la máquina (Fuente propia)

Después de haber instalado PyCharm, procederemos a ejecutarlo. Una vez lanzado el IDE, para poder seleccionar el intérprete Python que debe utilizar el programa, deberemos seleccionar la ruta de opciones "*File → New Project*". Tras ello, aparecerá una ventana donde deberemos indicar el intérprete base que se pretende usar, en este caso seleccionaremos la opción "Python 3.5", tal y como se puede observar en la figura 21.

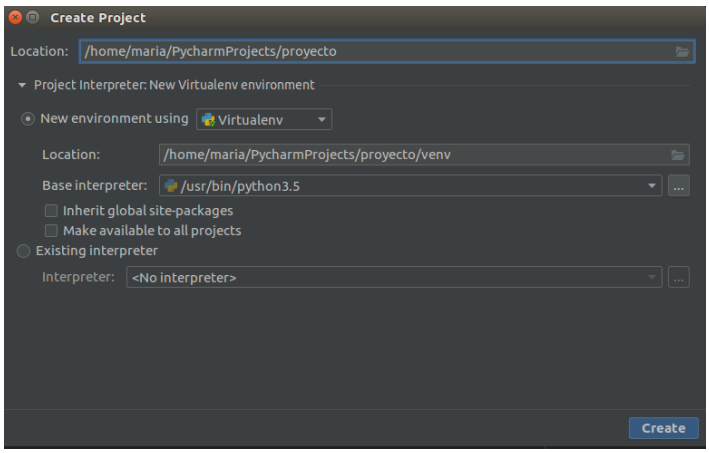


Figura 33. Captura con la versión de Pycharm instalada en la máquina (Fuente propia)

Así pues, ya tendremos el entorno configurado correctamente para poder comenzar el desarrollo.

6.1.3. Instalación de librerías con Pip

Para el desarrollo del algoritmo propuesto, será necesario añadir algunas librerías adicionales. Así pues, para ello inicialmente se necesitaremos instalar el sistema de gestión de paquetes de Python conocido con el nombre "Pip" para Python 3.X [55]. De esta manera, para instalarlo deberemos de ejecutar el siguiente comando sobre la terminal:

```
$ sudo apt-get install python3-pip
```

Una vez hecho esto, para comprobar que todo se ha instalado correctamente se procederá a revisar la versión de Pip que se tiene instalada en la máquina mediante el siguiente comando:

```
$ pip3 -V
```

Si todo ha salido correctamente, tras ejecutar dicho comando se deberá obtener como salida por pantalla la versión de Pip recién instalada, en este caso Pip 19.1.1.

Tras ello, una vez dispongamos del gestor de paquetes de Python, procederemos a instalar las dos librerías adicionales que necesitamos añadir: Tweepy [16] y Botometer [30]. Como bien se ha explicado en apartados anteriores, Tweepy será la librería que permitirá extraer los datos

referentes a cada una de las cuentas de usuario sujetas a estudio y, por otro lado, la librería Botometer permitirá valorar la cercanía que presente cada cuenta con un bot.

Así pues, para instalar ambas dependencias procederemos ejecutando los siguientes comandos utilizando para ello Pip:

```
$ pip3 install tweepy
```

```
$ pip3 install botometer
```

Una vez hecho esto, para comprobar que todo se han instalado ambas librerías correctamente se procederá a revisar la versión que se tiene instalada de cada una de ellas en la máquina mediante los siguientes comandos:

```
$ pip3 show tweepy
```

```
$ pip3 show botometer
```

Si todo ha salido correctamente, tras ejecutar dicho comando se deberá obtener como salida por pantalla las versiones recién instaladas de Tweepy y de Botometer, siendo en este caso Tweepy 3.8 y Botometer 1.3.

6.2. Extracción de la información a analizar

6.2.1. Llamada a la API y escritura

Inicialmente será necesario acceder a la API de Twitter, a través de la cual obtendremos los datos necesarios de cada perfil. Para ello, deberemos solicitar una cuenta de desarrollador en la plataforma Twitter y esperar a que se nos conceda.

Tras ello, una vez tengamos acceso a la cuenta de desarrollador, deberemos de dar de alta una "App" con el fin de que se asignen las claves de consumo de la API (secreta y pública) y los tokens de acceso (público y privado) que utilizaremos para poder acceder a los datos de Twitter [8].

En este caso, será necesario únicamente que hagamos uso de las claves de consumo de la API, tanto de la pública como de la secreta, creando para ello dos variables: "consumer_key" y "consumer_secret" respectivamente. Una vez hemos almacenado las claves proporcionadas por Twitter, haciendo uso de la librería Tweepy, se procede a crear una instancia de OAuthlander, a la que le deberemos de pasar ambas claves como parámetro, con el fin de que podamos acceder a la API de Twitter mediante la autenticación OAuth.

```
auth = tweepy.OAuthHandler(consumer_key,consumer_secret)
api = tweepy.API(auth)
```

Una vez se ha realizado correctamente la autenticación y se tiene acceso a los datos de la API, deberemos de indicar las cuentas de usuarios de las que deseamos obtener la información a estudiar. Para ello, se procede a parametrizar esta entrada de los usuarios escogidos en el código implementado, leyendo un archivo de extensión CSV donde se deberán indicar qué usuarios se van a consultar y los almacenaremos en una variable denominada "cuentas". Para esta lectura se hará uso de la librería o módulo de Python denominado "csv".

```
with open('/home/maria/PycharmProjects/tweepy/usuarios.csv') as origen
Datos:
      cuentas = csv.reader(origenDatos)
```

De esta manera, la disposición de las distintas cuentas dentro del fichero presenta la siguiente forma, en el caso del conjunto de datos de validación serán las 11 cuentas de twitter descritas:

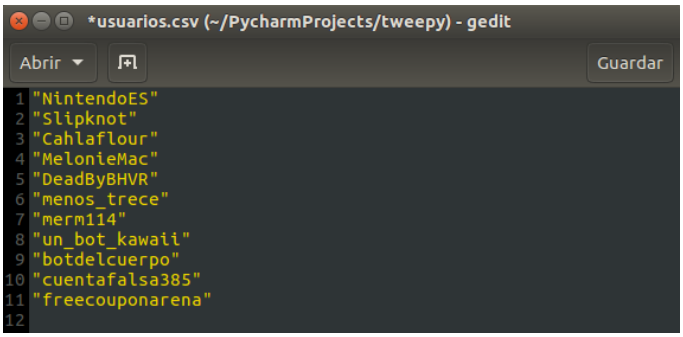


Figura 34. Captura del fichero CSV con los usuarios a buscar en la API.
(Fuente propia)

Una vez almacenadas las cuentas procedentes del fichero CSV, se procede a llamar a la API con el método "get_user" e indicaremos cada una de estas cuentas como parámetro, obteniendo asi el objeto completo del usuario con toda la información referente a su cuenta.

```
for cuenta in cuentas:
      user = api.get_user(cuenta[0])
```

Una vez separado cada objeto usuario ("user"), ya podremos llamar a los métodos concretos que nos permitirán obtener los parámetros que deseemos de cada cuenta.

```
# 1.Id
id = user.id

# 2.Seguidores
seguidores=user.followers_count
```

```
# 3.Amigos o seguidos
amigos=user.friends_count

# 4.Si la cuenta esta verificada o no
verificada=user.verified

# 5. Fecha de creacion de la cuenta
fecha_creacion=user.created_at

# 6. Descripcion
descripcion=user.description

# 7. Localizacion
localizacion=user.location

# 8. Imagen de perfil por defecto (el usuario no ha subido ninguna
imagen propia)
imagen_por_defecto=user.default_profile_image

# 9. Nombre del usuario
nombre=user.name

# 10. Numero de favoritos o me gustas
favoritos=user.favourites_count

# 11. Numero de publicaciones totales (propias + retuits)
publicaciones=user.statuses_count

# 12. Diseño de perfil por defecto (fondo y tema)
perfil_por_defecto=user.default_profile
```

Una vez obtenidos todos ellos, nos faltará el indicador referente a la puntuación de Botometer para cada usuario, el cual deberemos extraer utilizando la librería Botometer. Para ello, deberemos nuevamente de autenticar nuestro acceso a la API pasando las claves y tokens asignados a los métodos que ofrece Botometer. Una vez realizada la autenticación, se procederá a verificar cada una de las cuentas usando para ello el método "check_account" y se extraerá la puntuación que Botometer asigna para dicho usuario en base a todas sus características ("display_scores") independientes del idioma ("universal").

```
# 13. Nota bot (0-5)
taa = {
    'consumer_key': 'XXXXXXXXXXXXXX',
    'consumer_secret': ' XXXXXXXXXXXXXX',
    'access_token': ' XXXXXXXXXXXXXX',
    'access_token_secret': 'XXXXXXXXXXXXXX'
}

b = botometer.Botometer(wait_on_ratelimit=True, mashape_key="
XXXXXXXXXXXXXX", **taa)

datos_api = b.check_account('@'+cuenta[0])

nota_bot_universal = datos_api["display_scores"]["universal"]
```

Tras ello, una vez recogidos todos los datos necesarios de cada cuenta se procederá a escribirlos en un fichero CSV con el fin de proceder posteriormente a su análisis. Para ello, indicaremos un nombre para el fichero destino, el delimitador que se va a utilizar, el nombre de cada una de las columnas, así como los datos que se van a escribir en el mismo y la fecha y hora en la que se escribe esa medición.

```
cadena_datos = '/home/maria/PycharmProjects/tweepy/datos-usuarios.csv'

with open(cadena_datos,'a',newline='') as f_datos:

   writer_datos = csv.writer(f_datos,delimiter=';',quotechar='"')

   if os.stat(cadena_datos).st_size == 0: #primera fila fichero
      writer_datos.writerow(["columna1", "columna2",..., "columnaN"])

   writer_datos.writerow([dato1, dato2, ..., datoN])
```

6.2.2. Cron y Crontab

Cron resulta ser un administrador de procesos en segundo plano, el cual se encuentra presente en sistemas operativos Unix. Este, permite programar la ejecución de procesos en momentos o periodos de tiempo concretos. Esta especificación de cuando ejecutarlos o cada cuanto, se indica en un fichero denominado Crontab [56].

Este fichero Crontab, almacena en su interior las líneas de comandos que hemos especificado al administrador que se han de ejecutar de manera programada. Por su parte, Cron se encarga de ir revisando las líneas presentes en el fichero Crontab cada minuto, con el fin de comprobar si ha de ejecutar algún comando o tarea.

Así pues, para programar la ejecución automática del código que extrae los datos de cada usuario, se procede a hacer uso del fichero de texto Crontab. Este archivo, como se ha mencionado anteriormente, permite programar la ejecución de comandos en días, horas concretas y/o en intervalos regulares de tiempo.

En este caso, se programó para que ejecutara con Python 3 el archivo “indicadores.py”, con el fin de que recogiera los datos de los perfiles en intervalos de 24 horas y todos los días de la semana. Así pues, el fichero Crontab con esta ejecución programada, se muestra de la siguiente manera:

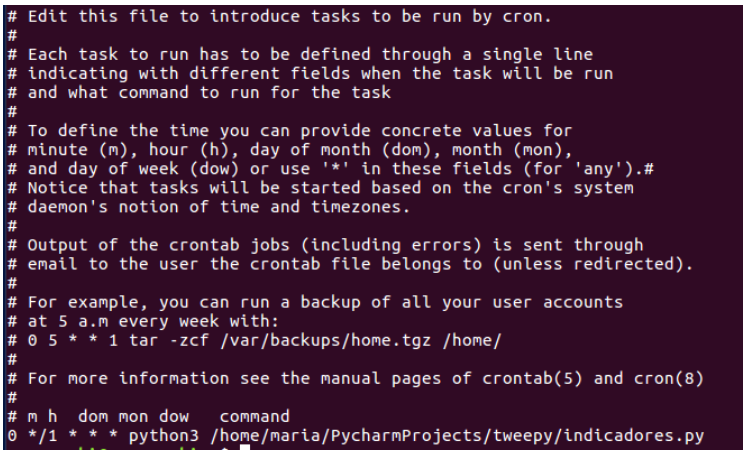


Figura 35. Captura del contenido del fichero Crontab
(Fuente propia)

6.2.3. Ejecución en paralelo

Con el fin de prevenir posibles pérdidas de datos ante causas inesperadas como actualizaciones o reinicios del sistema, se procedió a clonar la máquina virtual de este trabajo y a colocarla en dos ordenadores distintos. De esta manera, había dos equipos ejecutando simultáneamente la recogida de datos de los usuarios cada hora y, en caso de que uno se reiniciara o por la circunstancia que fuera dejara de lanzar el script, estaría el otro como respaldo.

De esta forma, tras finalizar la recogida de datos para esta primera prueba, se tenía como resultado 2 ficheros de extensión CSV con muchos datos duplicados entre sí, pues únicamente se diferenciaban por aquellos instantes concretos en los que cada ordenador pueda haber estado sin recoger datos.

Así pues, para eliminar estos duplicados y fusionar ambos ficheros, se procede a utilizar la herramienta Pentaho Data Integration. Se trata de una herramienta que permite múltiples operaciones relacionadas con el procesamiento de datos, mediante una interfaz intuitiva y usando el diseñador gráfico de extracción, transformación y carga (ETL) para ello [57].

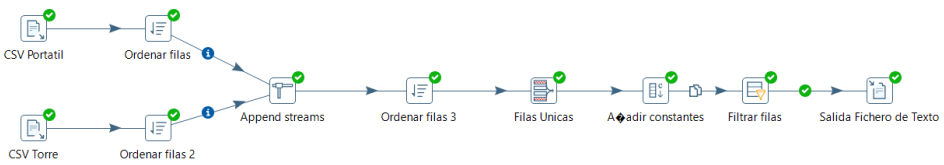


Figura 36. Captura de la ETL creada para unir los dos ficheros de datos
(Fuente propia)

De esta manera, el proceso ETL creado puede resumirse de la siguiente manera:

- Primeramente, se cargan y unen ambos orígenes de datos que previamente han sido ordenados (Append streams).
- Seguidamente, se procede a borrar todas aquellas filas duplicadas con el fin de dejar cada registro único, evitando los duplicados filtrando por cuenta de usuario y fecha (Filas Únicas).
- Tras ello, se filtran las filas para únicamente traer los datos referentes al mes que se va a estudiar (Filtrar filas).
- Finalmente, los datos de los usuarios procesados se vuelcan en un nuevo fichero de extensión CSV (Salida fichero).

6.3. Carga y procesamiento

6.3.1. Lectura de los indicadores

En primer lugar, se procede a leer el fichero resultado que hemos obtenido en el apartado anterior con los indicadores para cada una de las cuentas a estudiar, utilizando nuevamente la librería o módulo de Python denominado “csv”.

```
cadena_usuarios = '/home/maria/PycharmProjects/tweepy/datos-mes.csv'

with open(cadena_usuarios) as origenDatosUsuarios:

      usuarios = csv.reader(origenDatosUsuarios,delimiter=';')
```

Una vez recogidos todos los registros referentes a los indicadores de los usuarios, se procederá a separar cada uno de estos indicadores en variables distintas recorriendo para ello todas las filas leídas.

```
for usuario in usuarios:

  if usuario:

    if usuario[1] != "cuenta": #si no son nombres de columnas

      # datos cuenta usuario
      fecha_hora_medicion = usuario[0]
      nick_cuenta=usuario[1]
      id_cuenta=usuario[2]
      nombre_cuenta=usuario[3]
      retweets_diarios = int(usuario[14])
      tweets_diarios = int(usuario[13])
      seguidores = int(usuario[4])
      seguidos = int(usuario[5])
      fecha_creacion =
            datetime.strptime(usuario[7],'%Y-%m-%d %H:%M:%S')
      cadena = usuario[8]
      localizacion = usuario[9]
```

```
        perfil_defecto = usuario[15]
        likes = int(usuario[11])
        total_publis = int(usuario[12])
        botometer = float(usuario[16].replace(',','.'))
        verificada = usuario[6]
        imagen_por_defecto = usuario[10]
```

Con ello, ya disponemos de los indicadores de las cuentas que posteriormente serán necesarios para el cálculo de los factores determinantes y no determinantes.

6.3.2. Lectura del fichero de rangos

Los factores determinantes y no determinantes detallados en la propuesta refieren en todo momento a valores discretos y basados en aproximaciones que han sido sacadas de distintas fuentes. Por ello, en base a la posibilidad de que se quisiera cambiar en el futuro los valores de los rangos, las puntuaciones base y/o las puntuaciones de penalización para alguno de los factores, se ha decidido parametrizar estos datos, leyéndolos de un fichero con extensión CSV. En el caso de que para un factor no exista una segunda puntuación base, mitad de rango o cantidad a restar se colocará por defecto el valor 0.

	Predeterm	Predeterminado	Predeterminado	Predeterminado	Predeterminad	Predeterminado	Predetermin	Predeterminado	Predeterminado
1	factor	puntuacion_base	puntuacion_base2	inicio_rango	mitad_rango	mitad_rango2	fin_rango	cantidad_a_restar	cantidad_a_restar2
2	p_rd	100	0	0	0	0	10	50	100
3	p_td	100	0	0	0	0	20	50	100
4	p_su	100	0	0	0	0	707	50	100
5	p_rss	100	0	0	0.75	0	2	50	100
6	p_ac	100	0	0	0	0	1	100	0
7	p_dp	100	0	0	0	0	0	100	0
8	p_lp	100	0	0	0	0	0	100	0
9	p_pp	100	0	0	0	0	0	100	0
10	p_nl	100	0	0	0	0	200	50	100
11	p_ta	500	0	5	15	30	40	400	500
12	p_pb	500	0	1	0	0	5	500	0
13	p_vc	1000	1	0	0	0	0	0	0
14	p_ip	1	1	0	0	0	0	0	0

Figura 37. Captura del contenido del CSV donde se encuentran los valores para el cálculo de los factores (Fuente propia)

De esta manera, en el caso de que se quisiera cambiar alguno de estos valores discretos, únicamente sería necesario modificar este fichero CSV.

Seguidamente, una vez tengamos el fichero con los valores a utilizar para los factores, se procederá a leerlo utilizando, como en el caso de los indicadores, la librería o módulo de Python denominado “csv”.

```
cadena_rangos='/home/maria/PycharmProjects/tweepy/rangos_factores.csv'

with open(cadena_rangos) as origenDatosRangos:

        factores = csv.reader(origenDatosRangos)
```

Tras ello, se procede a almacenar en distintas variables los valores para rangos, puntuaciones base y puntuaciones de penalización presentes en cada registro, recorriendo para ello las distintas filas del fichero.

```
for factor in factores:
  nombre_factor = factor[0]

  if nombre_factor != "factor": #si no son nombres de columnas

    puntuacion_base = float(factor[1])
    puntuacion_base2 = float(factor[2])
    inicio_rango = float(factor[3])
    mitad_rango = float(factor[4])
    mitad_rango2 = float(factor[5])
    fin_rango = float(factor[6])
    cantidad_a_restar = float(factor[7])
    cantidad_a_restar2 = float(factor[8])
```

Una vez obtenidos estos valores y los asociados a los indicadores de cada cuenta, ya será posible proceder con el cálculo de las puntuaciones para cada uno de los factores determinantes y no determinantes.

6.3.3. Cálculo de los indicadores

Para el cálculo de la puntuación de los factores determinantes y no determinantes se ha definido una función individual para cada uno de los mismos. Así pues, a cada una de estas funciones se le pasarán los siguientes parámetros:

- Por un lado, estas funciones recibirán como parámetros las puntuaciones base, los valores límite de los rangos y los valores a restar leyendo para ello el fichero CSV donde se establece cada uno de ellos.
- Por otro lado, cada función además recibirá el valor o valores de las características de la cuenta que se han recogido de la API y que sean necesarios para el cálculo de la puntuación asociada a cada factor.

```
if nombre_factor == "p_rd":
  p_rd = ND_puntuacionRetweetsDiarios(puntuacion_base, inicio_rango,
  fin_rango, cantidad_a_restar, cantidad_a_restar2, retweets_diarios)

if nombre_factor == "p_td":
  p_td = ND_puntuacionTweetsDiarios(puntuacion_base, inicio_rango,
  fin_rango, cantidad_a_restar, cantidad_a_restar2, tweets_diarios)

if nombre_factor == "p_su":
  p_su = ND_puntuacionSeguidoresUsuario(puntuacion_base, inicio_rango,
  fin_rango, cantidad_a_restar, cantidad_a_restar2, seguidores)

if nombre_factor == "p_rss":
  p_rss = ND_puntuacionRatioSeguidoresSeguidos(puntuacion_base,
```

```
  inicio_rango, mitad_rango, fin_rango, cantidad_a_restar,
  cantidad_a_restar2, seguidores,seguidos)

if nombre_factor == "p_ac":
  p_ac = ND_puntuacionAntiguedadCuenta(puntuacion_base, inicio_rango,
  fin_rango, cantidad_a_restar, fecha_creacion)

if nombre_factor == "p_dp":
  p_dp = ND_puntuacionDescripcionPerfil(puntuacion_base,inicio_rango,
  fin_rango, cantidad_a_restar, cadena)

if nombre_factor == "p_lp":
  p_lp = ND_puntuacionLocalizacionPerfil(puntuacion_base,
  inicio_rango, fin_rango, cantidad_a_restar, localizacion)

if nombre_factor == "p_pp":
  p_pp = ND_puntuacionPersonalizacionPerfil(puntuacion_base,
  cantidad_a_restar, perfil_defecto)

if nombre_factor == "p_nl":
  p_nl = ND_puntuacionNumeroLikes(puntuacion_base, inicio_rango,
  fin_rango, cantidad_a_restar, cantidad_a_restar2, likes)

if nombre_factor == "p_ta":
  p_ta = ND_puntuacionPublicacionesAntiguedadDias(puntuacion_base,
  inicio_rango, mitad_rango, mitad_rango2, fin_rango,
  cantidad_a_restar, cantidad_a_restar2, total_publis, fecha_creacion)

if nombre_factor == "p_pb":
  p_pb = ND_puntuacionBotometer(puntuacion_base, inicio_rango,
  fin_rango, cantidad_a_restar,botometer)

if nombre_factor == "p_vc":
  p_vc = D_puntuacionVerificacionCuenta(puntuacion_base,
  puntuacion_base2, verificada)

if nombre_factor == "p_ip":
  p_ip = D_puntuacionImagenPerfil(puntuacion_base, puntuacion_base2,
  imagen_por_defecto)
```

6.3.4. Resultado final

Una vez ya se tienen las puntuaciones de todos y cada uno de los factores determinantes y no determinantes, se procede a calcular la puntuación final $P(CC_t)$. Para ello se seguirá la fórmula especificada en la propuesta, sumando los factores no determinantes en su valor absoluto (+1 para evitar que nunca pueda tomar valor 0) y, posteriormente, multiplicando esa cantidad por los factores determinantes.

```
puntuacion_total = round(abs(1+p_rd+p_td+p_su+p_rss+p_ac+p_dp+p_lp+p_p
p+p_nl+p_ta+p_pb)*(p_vc*p_ip),2)
```

6.3.5. Escritura de los resultados

Una vez calculado la puntuación final $P(CC_t)$ para cada una de las cuentas, se procederá a escribir los resultados en un fichero con extensión CSV. En este fichero únicamente guardaremos la fecha de medición, la cuenta del usuario y la puntuación final calculada.

```
cadena_escritura = '/home/maria/PycharmProjects/tweepy/resultados.csv'

with open(cadena_escritura, 'a', newline='') as f_datos:
      writer_datos = csv.writer(f_datos, delimiter=';', quotechar='"')

      if os.stat(cadena_escritura).st_size == 0:

            writer_datos.writerow(["fecha_hora_medicion", "cuenta",
            "puntuacion_total"])
            ...

            writer_datos.writerow([fecha_hora_medicion, nick_cuenta,
            puntuacion_total])
```

Así pues, si todo ha salido correctamente una vez ejecutemos el código obtendremos el fichero CSV con las puntuaciones finales de cada usuario con la siguiente apariencia:

fecha_hora_medicion	cuenta	puntuacion_total
19/07/2019	botdelcuerpo	751
19/07/2019	Cahlaflour	801
19/07/2019	cuentafalsa385	151
19/07/2019	DeadByBHVR	801000
19/07/2019	freecouponarena	501
19/07/2019	MelonieMac	1301
19/07/2019	menos_trece	1151000
19/07/2019	merm114	151
19/07/2019	NintendoES	651000
19/07/2019	Slipknot	1301000
19/07/2019	un_bot_kawaii	801

Figura 38. Captura de un fragmento del contenido del CSV donde se han escrito las puntuaciones finales calculadas.
(Fuente propia)

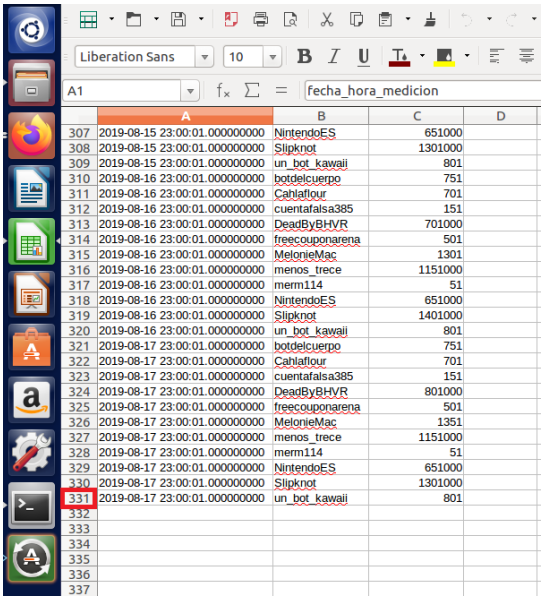
Estas operaciones se repetirán para cada día de toma de datos, de forma que ya sea una semana o un mes, tendremos el cálculo de la puntuación para cada cuenta.

6.4. Validación de la implementación

Para validar la implementación del código realizado, el experimento que realizaremos será utilizar el código con las 11 cuentas de validación utilizadas para validad el algoritmo. La idea de este experimento es comprobar que el código produce los mismos resultados que la captura de datos manual que se realizó durante la fase de validación propuesta. Esto nos permitirá

posteriormente poder realizar pruebas conociendo fehacientemente que el código está libre de errores.

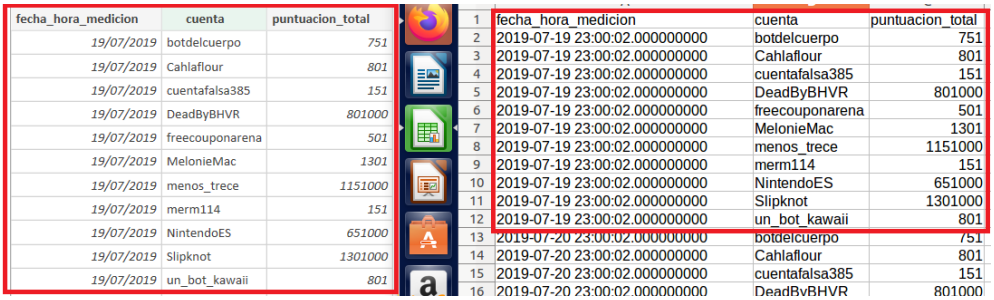
En este caso puesto que se trataba de los mismos 11 usuarios que los tratados en la propuesta y para el conjunto de los 30 días de monitorización, se deberá de haber obtenido en este fichero de salida un total de 330 registros (descontando la fila con el nombre de las columnas, sino sería un total de 331). Así pues, si comprobamos el número de filas del fichero obtenido como salida de la ejecución del algoritmo desarrollado, podemos confirmar que efectivamente la cuantía obtenida coincide con la esperada según la propuesta planteada con anterioridad, tal y como se puede observar en la siguiente imagen:



	A	B	C	D
307	2019-08-15 23:00:01.000000000	NintendoES	651000	
308	2019-08-15 23:00:01.000000000	Slipknot	1301000	
309	2019-08-15 23:00:01.000000000	un_bot_kawaii	801	
310	2019-08-16 23:00:01.000000000	botdelcuerpo	751	
311	2019-08-16 23:00:01.000000000	Cahlaflour	701	
312	2019-08-16 23:00:01.000000000	cuentafalsa385	151	
313	2019-08-16 23:00:01.000000000	DeadByBHVR	701000	
314	2019-08-16 23:00:01.000000000	freecouponarena	501	
315	2019-08-16 23:00:01.000000000	MelonieMac	1301	
316	2019-08-16 23:00:01.000000000	menos_trece	1151000	
317	2019-08-16 23:00:01.000000000	merm114	51	
318	2019-08-16 23:00:01.000000000	NintendoES	651000	
319	2019-08-16 23:00:01.000000000	Slipknot	1401000	
320	2019-08-16 23:00:01.000000000	un_bot_kawaii	801	
321	2019-08-17 23:00:01.000000000	botdelcuerpo	751	
322	2019-08-17 23:00:01.000000000	Cahlaflour	701	
323	2019-08-17 23:00:01.000000000	cuentafalsa385	151	
324	2019-08-17 23:00:01.000000000	DeadByBHVR	801000	
325	2019-08-17 23:00:01.000000000	freecouponarena	501	
326	2019-08-17 23:00:01.000000000	MelonieMac	1351	
327	2019-08-17 23:00:01.000000000	menos_trece	1151000	
328	2019-08-17 23:00:01.000000000	merm114	51	
329	2019-08-17 23:00:01.000000000	NintendoES	651000	
330	2019-08-17 23:00:01.000000000	Slipknot	1301000	
331	2019-08-17 23:00:01.000000000	un_bot_kawaii	801	
332				
333				
334				
335				
336				
337				

Figura 39. Captura que muestra las 330 filas generadas con las puntuaciones $P(CC_t)$ para cada usuario. (Fuente propia)

Además, se procede a comparar las puntuaciones totales calculadas manualmente del apartado de la propuesta con los resultados obtenidos para la puntuación total tras la ejecución del algoritmo, con el fin de verificar que las puntuaciones para cada usuario reflejadas para el día 19/07/2019 concuerdan en ambos casos. Así pues, si observamos ambos ficheros se puede corroborar que los resultados coinciden, tal y como se puede apreciar en la siguiente imagen:



fecha_hora_medicion	cuenta	puntuacion_total
19/07/2019	botdelcuerpo	751
19/07/2019	Cahlaflour	801
19/07/2019	cuentafalsa385	151
19/07/2019	DeadByBHVR	801000
19/07/2019	freecouponarena	501
19/07/2019	MelonieMac	1301
19/07/2019	menos_trece	1151000
19/07/2019	merm114	151
19/07/2019	NintendoES	651000
19/07/2019	Slipknot	1301000
19/07/2019	un_bot_kawaii	801

	fecha_hora_medicion	cuenta	puntuacion_total
1	fecha_hora_medicion	cuenta	puntuacion_total
2	2019-07-19 23:00:02.000000000	botdelcuerpo	751
3	2019-07-19 23:00:02.000000000	Cahlaflour	801
4	2019-07-19 23:00:02.000000000	cuentafalsa385	151
5	2019-07-19 23:00:02.000000000	DeadByBHVR	801000
6	2019-07-19 23:00:02.000000000	freecouponarena	501
7	2019-07-19 23:00:02.000000000	MelonieMac	1301
8	2019-07-19 23:00:02.000000000	menos_trece	1151000
9	2019-07-19 23:00:02.000000000	merm114	151
10	2019-07-19 23:00:02.000000000	NintendoES	651000
11	2019-07-19 23:00:02.000000000	Slipknot	1301000
12	2019-07-19 23:00:02.000000000	un_bot_kawaii	801
13	2019-07-20 23:00:02.000000000	botdelcuerpo	751
14	2019-07-20 23:00:02.000000000	Cahlaflour	801
15	2019-07-20 23:00:02.000000000	cuentafalsa385	151
16	2019-07-20 23:00:02.000000000	DeadByBHVR	801000

Figura 40. Captura que muestra a la izquierda la puntuación total calculada de manera manual y a la derecha la puntuación total obtenida tras la ejecución del algoritmo implementado. (Fuente propia)

Así pues, como el código implementado produce los resultados esperados podemos decir que es válido y no presenta errores de programación. De esta manera, se puede afirmar que el proceso de obtención de los datos de cada usuario de Twitter a través de la API, el cálculo de las puntuaciones asociadas a cada factor determinante y no determinante, el cálculo de la puntuación final, así como la ejecución en paralelo del algoritmo en dos ordenadores para evitar posibles pérdidas de datos y la fusión de ambos ficheros resultantes para unificar toda la información, se consideran procedimientos correctos y totalmente funcionales.

7. Prueba de la propuesta

En este apartado se procede a utilizar el algoritmo desarrollado para realizar una prueba sobre un gran conjunto de cuentas de usuarios, con el fin de estudiar los resultados finales, contrastar su validez y ver qué valor o importancia aportan estas puntuaciones calculadas.

En este caso se pretende estudiar el perfil de usuario medio que presentan entre sus seguidores algunos partidos políticos en su cuenta de la red social Twitter. Ya que este TFG ha coincidido en el tiempo con el escenario de las elecciones electorales del 10 de noviembre, considero que es un buen momento para analizar y estudiar qué resultados puede proporcionarnos este estudio de la calidad de los seguidores que tiene cada partido político.

Así pues, para realizar estas pruebas se procederá a seleccionar un grupo de N cuentas de Twitter de entre los seguidores de cada partido, con el fin de determinar el $P(CC_t)$ medio que presentan los seguidores de cada agrupación política, es decir, en primer lugar se calculará el $P(CC_t)$ de cada usuario y tras ello, se procederá al cálculo del $P(CC_t)$ medio los seguidores del partido, sobre el cual se estudiará su tendencia en el tiempo, con el fin de determinar cómo de buenas o malas son las cuentas de usuarios que siguen a cada agrupación en la red social, analizando de esta manera su grado de credibilidad.

De esta forma, la media de las puntuaciones finales de las cuentas se llevará a cabo realizando la media aritmética ya que, aunque hay otras formas y/o métodos de evaluación, se ha escogido esta forma con el fin de no engrosar la complejidad del análisis que se va a realizar.

7.1. Elección del conjunto de cuentas

Aprovechando el escenario de las elecciones generales del 10 de noviembre, se procede a realizar un análisis relacionado con este contexto político. Para ello, se seleccionarán 100 cuentas de seguidores que tengan 4 partidos políticos, sumando un total de 400 usuarios. Esta selección se llevará a cabo visitando los perfiles de cada partido implicado visualizando sus seguidores y seleccionando los primeros 100 usuarios que aparezcan en esa lista.

Las cuentas de partidos políticos con las que se van a trabajar son 4, donde 2 pertenecen a partidos tradicionales (PSOE y PP) y las otras 2 pertenecen a 2 partidos más recientes y polémicos (Podemos y Vox). Así pues, las cuentas oficiales de Twitter para cada partido son las siguientes:

- Cuenta oficial de Twitter del PSOE: @PSOE
- Cuenta oficial de Twitter de PP: @populares
- Cuenta oficial de Twitter de Podemos: @ahorapodemos
- Cuenta oficial de Twitter de Vox: @vox_es

Las listas de los seguidores extraídos de la cuenta de cada partido político se adjuntan en el anexo I de este trabajo.

7.2. Procesamiento y cálculo de datos

Se procede a extraer los datos de cada una de las cuentas de los seguidores de los partidos políticos, con el fin de calcular los factores determinantes y no determinantes y, posteriormente, calcular la puntuación final. En este caso, se va a trabajar con 400 cuentas (100 seguidores de cada partido) y se va a recopilar datos de dichos usuarios durante un periodo de 7 días (desde el día 6 de noviembre hasta el día 12 de noviembre inclusive). De igual forma que en el caso de las cuentas utilizadas para validar la propuesta e implementación, se recogerá la información de los usuarios a las 23:00 de cada día.

Así pues, 400 cuentas de usuarios recogiendo 1 registro de datos por día de cada una de las mismas durante un periodo de 7 días indica que se habrá de obtener un total de 2800 registros, tanto de datos a procesar provenientes de la API como de puntuaciones finales $P(CC_t)$ calculadas. Además, puesto que se trata de un número notorio de cuentas a analizar agrupadas por partido político, se va a realizar la media aritmética de cada grupo de seguidores, con el fin de estudiar la tendencia del $P(CC_t)$ medio de los seguidores de cada partido político.

Así pues, la gráfica resultante con las tendencias de los 4 partidos políticos resulta ser la siguiente:

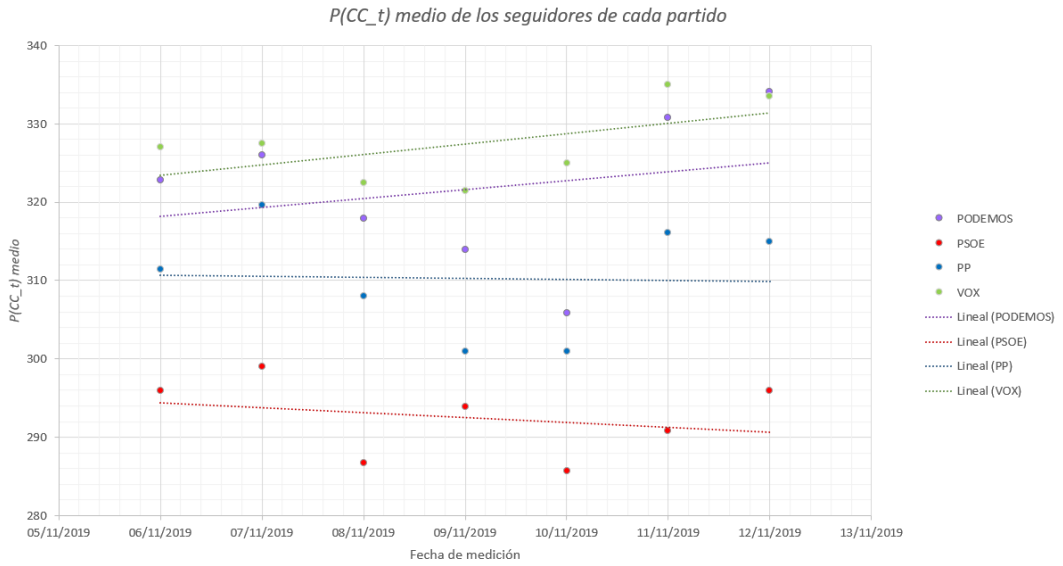


Figura 41. Captura de la gráfica resultante de la tendencia del $P(CC_t)$ medio de los seguidores de cada partido durante los 7 días de medición.
(Fuente propia)

7.3. Evaluación de resultados

En este apartado se procede a realizar la evaluación de los resultados obtenidos sobre el $P(CC_t)$ medio que presentan los seguidores de cada uno de los partidos políticos implicados en el estudio.

PSOE

Si analizamos la gráfica obtenida para esta cuenta, observamos una tendencia lineal que decrece en el tiempo. Aunque se observan valores muy dispares para los 7 días de medición, llama la atención el $P(CC_t)$ medio que se incrementa ligeramente el día de antes de las elecciones y sin embargo, al día siguiente, esta puntuación decrece casi en 20 puntos.

Por un lado, la subida del $P(CC_t)$ medio el día de antes podría ser debido a que al ser el último día antes de las elecciones se puede producir un aumento de actividad entre sus seguidores, ya que es muy habitual que aumente la polémica en las redes sociales en fechas previas a un evento así. Por otra parte, el hecho de que disminuya la puntuación el mismo día que se celebran las elecciones puede ser síntoma de que al haber tenido lugar ya el acontecimiento, la actividad polémica en la red social disminuye y ya no vuelve a recuperarse hasta el día siguiente que es cuando los usuarios ya conocen los resultados de las elecciones en su gran mayoría. Además,

otras opciones a barajar como posibles causas de esta bajada de puntuación tan notoria, podría ser también debido a la presencia de bots o cuentas falsas.

Así pues, como valoración final para estas cuentas de seguidores del PSOE se les otorga la categoría de "cuentas malas", puesto que como "cuentas buenas" entendemos aquellas que más o menos permanecen estables en el tiempo y esta, por el contrario, presenta una tendencia en su puntuación final media que va decreciendo con el transcurso de los días.

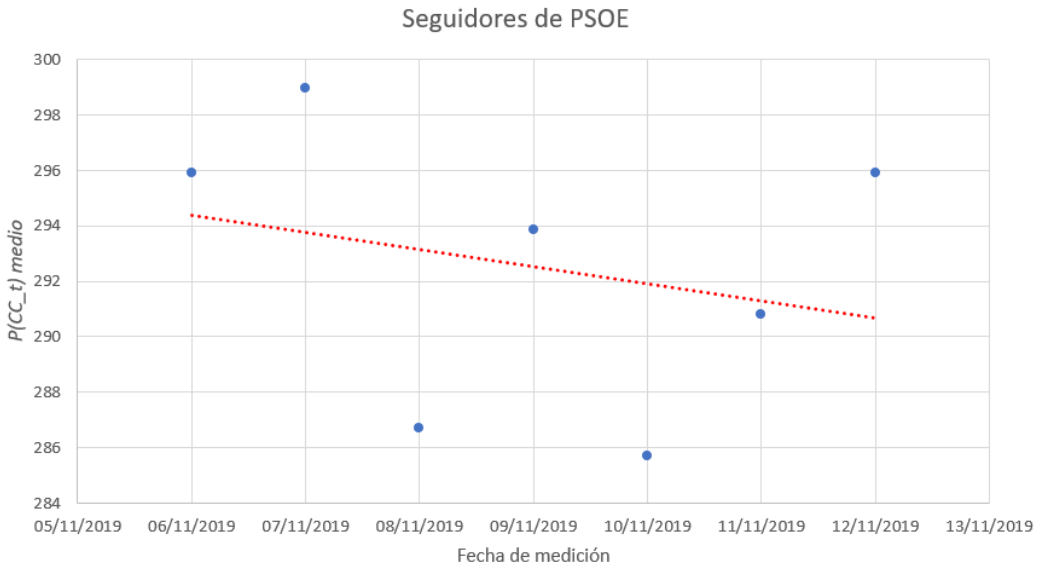


Figura 42. Captura de la gráfica resultante de la tendencia del $P(CC_t)$ medio de los seguidores del partido PSOE.
(Fuente propia)

PP

Si analizamos la gráfica obtenida para esta cuenta, observamos una tendencia lineal que se mantiene más o menos constante en el tiempo, aunque presenta un ligero decrecimiento. Aunque se observan valores muy dispares para los 7 días de medición, llama enormemente la atención el $P(CC_t)$ medio tan bajo que se recoge para los días 9 y 10 de noviembre. Este hecho podría haberse visto afectado, al igual que en el caso anterior, por factores como la repercusión mediática que supone un aumento en la actividad de los usuarios en la red social los días anteriores a las elecciones o después de las mismas, pues el mismo día de las elecciones puede entrar dentro de lo normal que la actividad baje ligeramente hasta conocer los resultados.

Este decrecimiento en la tendencia del $P(CC_t)$ medio para los seguidores de este partido no es tan importante como en el caso anterior, pues supone una bajada de puntuación final apenas perceptible. Por tanto, a este conjunto de seguidores se le otorga la categoría de "cuentas buenas", puesto que su tendencia como bien se ha comentado anteriormente se mantiene más

o menos constante en el tiempo de medición, hecho que indica que sus usuarios presentan la misma actividad, sin picos ni sobresaltos extraños.

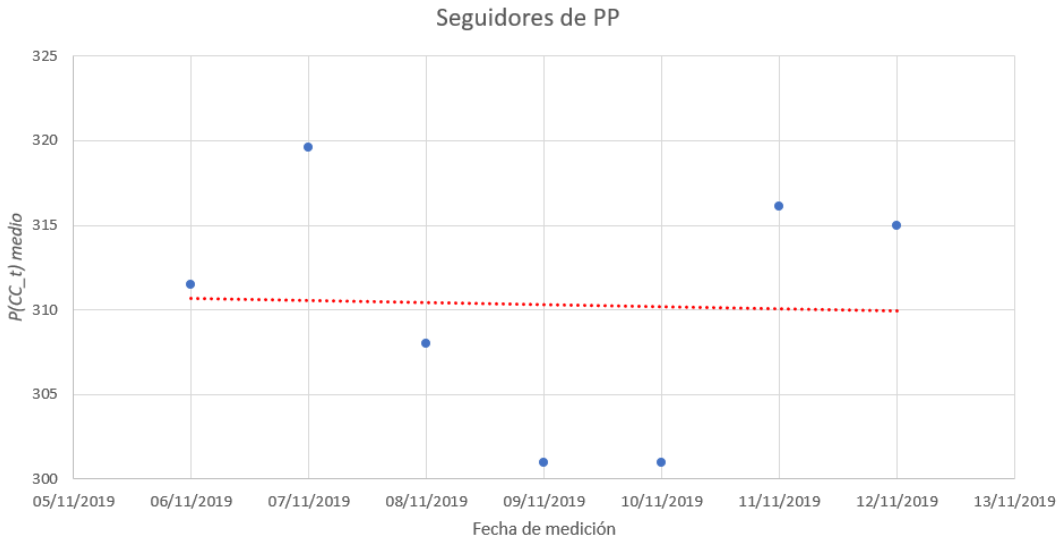


Figura 43. Captura de la gráfica resultante de la tendencia del $P(CC_t)$ medio de los seguidores del partido PP.
(Fuente propia)

Podemos

Si analizamos la gráfica obtenida para esta cuenta, observamos una tendencia lineal que se mantiene más o menos constante en el tiempo, aunque presenta un ligero crecimiento. Nuevamente, encontramos una bajada importante del $P(CC_t)$ medio durante el 10 de noviembre, día de las elecciones. Sin embargo, si observa que este decrecimiento ha sido paulatino, es decir, no ha sido de un día para otro.

Por otra parte, si analizamos la puntuación final media obtenida para los días posteriores a las elecciones, podemos observar que la puntuación de los usuarios ha aumentado bastante, superando el valor máximo recogido con anterioridad para el día 7 de noviembre. Este hecho podría deberse a que, ya que se trata de un partido novedoso y bastante reciente, por norma general despierta y genera una mayor polémica entre los usuarios de la red social. Al tratarse de una subida en la tendencia lineal muy ligera, entra dentro de lo normal que se podría esperar de una cuenta que está creciendo de manera correcta sin presentar valores atípicos que generen sospechas. Así pues, considero que la categoría para los seguidores de esta cuenta debería de ser "cuentas buenas", ya que la tendencia de $P(CC_t)$ medio apenas varía y lo hace de manera muy discreta y de manera creciente, entrando dentro de márgenes normales en el crecimiento sano y normal de cuentas de usuarios.

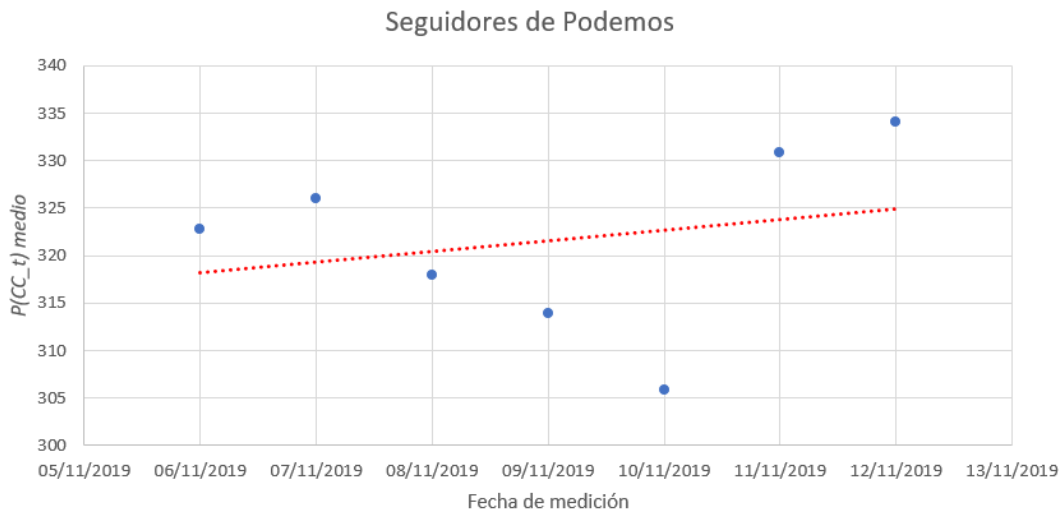


Figura 44. Captura de la gráfica resultante de la tendencia del $P(CC_t)$ medio de los seguidores del partido Podemos.
(Fuente propia)

Vox

Si analizamos la gráfica obtenida para esta cuenta, observamos una tendencia lineal que presenta un crecimiento bastante acentuado. En este caso, a diferencia de los anteriores, el día 10 de noviembre se observa un ligero crecimiento del $P(CC_t)$ medio asociado a los seguidores de dicha cuenta, mientras que para los días tras las elecciones se produce un aumento muy notorio de la puntuación final de sus seguidores. Este hecho podría deberse a que, al igual que en el caso anterior, al tratarse de un partido novedoso y bastante reciente, normalmente genera mayor polémica y este hecho puede ser un factor de relevancia que aumente enormemente la actividad de sus seguidores. Sin embargo, una tendencia con un crecimiento tan notorio resulta raro, ya que lo habitual en un crecimiento sano y normal de una cuenta es un crecimiento paulatino, no exponencial. Así pues, este hecho podría llevar a pensar en que dichos resultados sean derivados de alguna campaña táctica y por tanto, la categoría que se le asigna a las cuentas de los seguidores de este partido resulta ser la de “cuentas malas”.

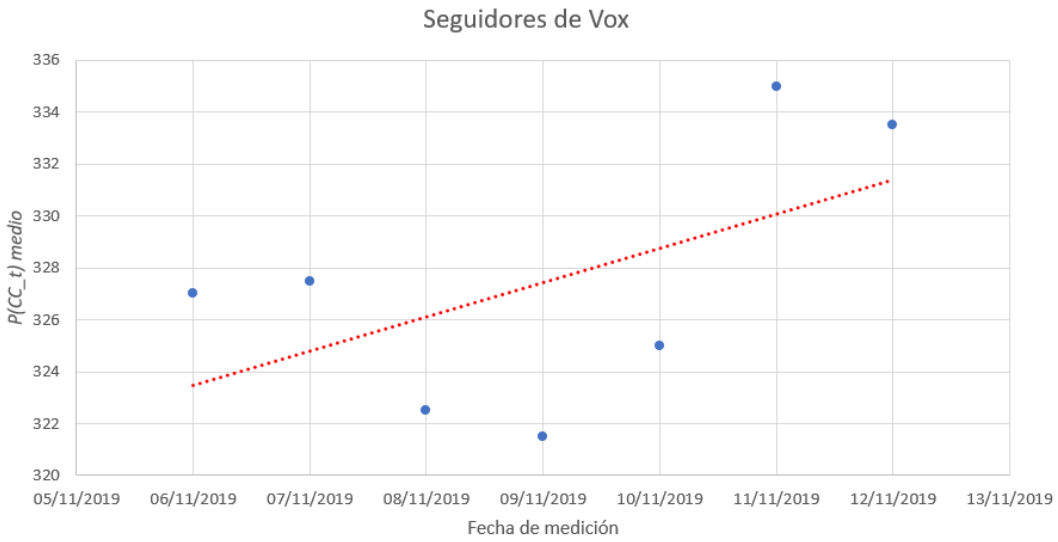


Figura 45. Captura de la gráfica resultante de la tendencia del $P(CC_t)$ medio de los seguidores del partido Vox.
(Fuente propia)

8. Conclusiones y trabajo futuro

Como conclusión sobre el trabajo realizado, se puede extraer el hecho de que el algoritmo desarrollado cobra firmeza y validez si se aplica a datos de usuarios recogidos en un periodo de tiempo amplio, pues es realmente la tendencia que presentan sus puntuaciones finales lo que ayuda a determinar si es mejor o peor dicha cuenta. Los resultados obtenidos en las pruebas nos permiten comprobar que es esta propuesta es un posible mecanismo de clasificación pero que aún le queda mucho por mejorar para lograr unos resultados mucho más certeros y objetivos. Sin embargo el objetivo general de este TFG ha sido cubierto, el de proponer un algoritmo formal basado factores certeros de los que depende la credibilidad de las cuentas en la red social Twitter. Como mayor debilidad podemos notar que cuando las cuentas son hábilmente manipuladas puede ser que la credibilidad pueda esconderse, mientras que en el caso de cuentas de tipo bot, latientes, usuarios falsos, con este algoritmo pueden ser fácilmente detectadas.

Creo que este proyecto es un paso más para colaborar en la detección de perfiles falsos, poco fiables y/o controlados por algún tipo de software, sobretodo en épocas determinadas que pueden propiciar la aparición de perfiles falsos como es el caso de las elecciones electorales para simular que un partido presenta un mayor número de seguidores.

Además, cabe mencionar que el algoritmo propuesto presenta una complejidad lineal en su coste computacional, pues únicamente depende del tamaño del conjunto de datos que se le pase a analizar.

El algoritmo propuesto presenta múltiples posibles formas de mejora o posibles correcciones, algunas de estas posibles mejoras de cara a un trabajo futuro podrían ser las siguientes:

- Utilización de valores continuos en lugar de discretos, con el fin de afinar aún más las puntuaciones obtenidas para cada usuario.
- Incrementar el número de factores determinantes y no determinantes, así como repensar si los existentes que actualmente se utilizan son relevantes o si de lo contrario alguno no aporta un peso significativo y por tanto se puede prescindir del mismo.
- Generar un algoritmo de clasificación automático capaz de recoger los distintos factores que influyen en la credibilidad de una cuenta como son: la tendencia media de valor de la cuenta, la variación de los valores de la cuenta, los valores mínimos y máximos, margen de error, etc. Además, este algoritmo podría estar basado en diferentes técnicas

como redes neuronales o machine learning que permitan un autoaprendizaje en función de la experiencia, lo cual pueda además aprovechar los datos que se van recogiendo para afinar más la clasificación.

- Estudiar el lenguaje presente en cada uno de los tweets de los usuarios en el tiempo en que se monitorice cada cuenta, con el fin de afinar con mayor profundidad el hecho de que se trate de una cuenta perteneciente a una persona real o bien, se trate de un perfil controlado por algún tipo de software. Para ello se podrían analizar cosas tales como cantidad de palabras que expresen sentimiento u opinión en sus publicaciones, así como los contextos de palabras que presentan sus publicaciones.
- Estudiar la repercusión que tienen las publicaciones de un usuario en base a la calidad de su contenido midiendo el número de retweets y favoritos que este presenta, con el fin de acabar con los falsos positivos que actualmente no controla el algoritmo propuesto.
- Estudiar las horas y periodicidad con las que cada cuenta efectúa publicaciones, ya que muchos bots publican tweets en intervalos concretos de tiempo como cada hora, cada 15 minutos, etc. Normalmente, las personas que hacen uso de la red social Twitter no realizan publicaciones en horas concretas coincidentes todos los días o en intervalos exactos de tiempo, más bien sus publicaciones están sujetas a la aleatoriedad de cuando al usuario le apetece escribir un tweet y compartirlo.

Referencias

1. BBC. *WhatsApp restricts message-sharing to fight fake news.* 2019. Disponible en: https://www.bbc.com/news/technology-46945642
2. BBC. *Facebook employs UK fact-checkers to combat fake news.* 2019. Disponible en: https://www.bbc.com/news/technology-46836897
3. Dwoskin, E. Experiencia de Usuario: corresponsal de Silicon Valley en The Washington Post. *Facebook is rating the trustworththiness of its users on a scale from zero to 1.* 2018. Disponible en: https://www.washingtonpost.com/technology/2018/08/21/facebook-is-rating-trustworthiness-its-users-scale-zero-one/?noredirect=on&utm_term=.9d0a2672f961
4. Shyrokykh, K. Experiencia de Usuario: investigadora estadística en Ericsson Consumer & IndustryLab sobre la industria de las TIC. *Fake new son social media: Whose responsibility is it?* 2018. Disponible en: https://www.ericsson.com/en/blog/2018/11/fake-news-on-social-media-whose-responsibility-is-it
5. McCarthy, N. Experiencia de Usuario: periodista de datos en Statista sobre temas tecnológicos, sociales y de medios visuales. *Where Exposure To Fake News Is Highest [Infographic].* 2018. Disponible en: https://www.forbes.com/sites/niallmccarthy/2018/06/14/where-exposure-to-fake-news-is-highest-infographic/
6. Documentación guía sobre Twitter. Disponible en: https://help.twitter.com/en/twitter-guide
7. Documentación sobre la API de Twitter. Disponible en: https://help.twitter.com/es/rules-and-policies/twitter-api
8. Documentación sobre la autenticación y autorización de las API de Twitter. Disponible en: https://developer.twitter.com/en/docs/basics/authentication/overview/authentication-and-authorization
9. Documentación sobre la autenticación básica de las API empresariales de Twitter. Disponible en: https://developer.twitter.com/en/docs/basics/authentication/overview/basic-auth
10. Documentación sobre la autenticación OAuth. Disponible en: https://OAuth.net/articles/authentication/

11. Documentación sobre el modelo de autenticación OAuth de las API de Twitter. Disponible en: https://developer.twitter.com/en/docs/basics/authentication/overview/OAuth
12. The Economist. *Python is becoming the world's most popular coding language.* 2018. Disponible en: https://www.economist.com/graphic-detail/2018/07/26/python-is-becoming-the-worlds-most-popular-coding-language
13. Feldman, S. Experiencia de Usuario: periodista de datos en Statista. *The Most Popular Programming Languages.* 2019. Disponible en: https://www.statista.com/chart/16567/popular-programming-languages/
14. Documentación sobre las librerías creadas para la plataforma de Twitter. Disponible en: https://developer.twitter.com/en/docs/developer-utilities/twitter-libraries
15. Documentación sobre la autenticación usando la librería Tweepy. Disponible en: https://tweepy.readthedocs.io/en/3.7.0/auth_tutorial.html
16. Documentación sobre la clase principal de Tweepy y sus métodos principales. Disponible en: https://tweepy.readthedocs.io/en/latest/api.html
17. Documentación sobre la autenticación usando la librería Twython. Disponible en: https://twython.readthedocs.io/en/latest/usage/starting_out.html#authentication
18. Documentación sobre la clase principal Twython y sus métodos principales. Disponible en: https://twython.readthedocs.io/en/latest/api.html
19. Documentación sobre la autenticación usando la librería TwitterAPI. Disponible en: http://geduldig.github.io/TwitterAPI/authentication.html
20. Documentación sobre la clase principal TwitterAPI y su método request. Disponible en: https://github.com/geduldig/TwitterAPI
21. Amengual, J. Experiencia de Usuario: consultor de marketing y ventas, experto en marketing online y social media marketing. *Las estadísticas de las redes sociales.* Disponible en: https://www.marketingproductivo.es/las-estadisticas-las-redes-sociales/
22. Lohr, S. Experiencia de Usuario: reportero sobre tecnología y economía en The New York Times. *It's True: False News Spreads Faster and Wider. And Humans Are to Blame.* Disponible en: https://www.nytimes.com/2018/03/08/technology/twitter-fake-news-research.html
23. Documentación sobre Twitter Analytics para empresas. Disponible en: https://business.twitter.com/es/analytics.html
24. Gupta, A. (Indraprastha Institute of Information Technology Delhi, India), Lamba, H. (IBM, IRL, New Delhi, India), Kumaraguru, P. (Indraprastha Institute of Information

Technology, New Delhi, India) y Joshi, A. (University of Maryland Baltimore County, Baltimore, MD, USA). *Faking Sandy: Characterizing and Identifying Fake Images on Twitter during Hurricane Sandy.* Disponible en: https://mdsoar.org/bitstream/handle/11603/11856/677.pd.pdf?sequence=1&isAllowed=y

25. Conroy, N. (University of Western Ontario), Rubin, V. (Associate Prof., Faculty of Information & Media Studies, Language & Info. Tech. Lab, Western) y Chen, Y. (University of Western Ontario). *Automatic Deception Detection: Methods for Finding Fake News.* Disponible en: https://onlinelibrary.wiley.com/doi/epdf/10.1002/pra2.2015.145052010082
26. Volkova, S. (Pacific Northwest National Laboratory), Shaffer, K. (Qntfy), Jang, JY. (Korea Electronics Technology Institute) y Hodas, N. (Pacific Northwest National Laboratory). *Separating Facts from Fiction: Linguistic Models to Classify Suspicious and Trusted News Posts on Twitter.* Disponible en: http://www.aclweb.org/anthology/P17-2102
27. *S*ao, C. (National University of Defense Technology), Ciampaglia GL. (University of South Florida), Varol, O. (Postdoctoral Research Associate at Northeastern University). Flammini, A. (Indiana University - School of Informatics and Computing) y Menczer, F. (Professor of Informatics and Computer Science, Indiana University). *The spread of fake news by social bots.* Disponible en: https://www.andyblackassociates.co.uk/wp-content/uploads/2015/06/fakenewsbots.pdf
28. Documentación sobre la herramienta Botometer. Disponible en: https://botometer.iuni.iu.edu/#!/
29. Preguntas frecuentes sobre la herramienta Botometer. Disponible en: https://botometer.iuni.iu.edu/#!/faq
30. Documentación sobre la API de Botometer. Disponible en: https://botometer.iuni.iu.edu/#!/api
31. Meneses Berastegui, G. (Socio, director creativo y estratega de Digital Bakers). *Las fake news: quién las crea, para qué y cómo se propagan.* 2018. Disponible en: http://www.uncode.cafe/mentiras-y-verdades/las-fake-news-quien-las-crea-para-que-y-como-se-propagan
32. Garzón, B. Experiencia de usuario: Jurista y promotor de Actúa. *Fake news.* 2019. https://www.eldiario.es/tribunaabierta/Fake-news_6_883021725.html
33. Documentación sobre Twitter Ads. Disponible en: https://ads.twitter.com
34. Documentación sobre casos de éxito de los servicios que ofrece Twitter para empresas. Disponible en: https://business.twitter.com/es/success-stories.html

35. Marketing directo. *Las claves para twittear con éxito y credibilidad*. 2012. Disponible en: https://www.marketingdirecto.com/digital-general/social-media-marketing/las-claves-para-twittear-con-exito-y-credibilidad?utm_source=wordtwit&utm_medium=social&utm_campaign=wordtwit
36. NUSGREM (Asociación Nacional de Estudiantes Universitarios de Ciencias Físicas). *Información y entropía, un acercamiento probabilístico.* 2017. Disponible en: https://nusgrem.es/informacion-entropia-probabilidad/
37. Gurajala, S. (Department of Computer Science, Clarkson University, USA). *Profile characteristics of fake Twitter accounts.* 2016. Disponible en: https://journals.sagepub.com/doi/full/10.1177/2053951716674236
38. Documentación oficial sobre las cuentas verificadas de Twitter. Disponible en: https://help.twitter.com/es/managing-your-account/about-twitter-verified-accounts
39. Documentación M., Y. (Editor de Xataka en Webedia España y responsable de Xataka Basics). *Qué es una cuenta verificada de Twitter.* 2017. Disponible en: https://www.xataka.com/basics/que-es-una-cuenta-verificada-de-twitter
40. Documentación oficial de Twitter sobre las preguntas frecuentas en relación con las cuentas verificadas. Disponible en: https://help.twitter.com/es/managing-your-account/twitter-verified-accounts
41. Rubio, J. (Redactor en el periódico El País - Verne). *Twitter se queda sin huevos.* 2017. Disponible en: https://verne.elpais.com/verne/2017/04/03/articulo/1491206749_017669.html
42. Enfroy, A. (Experto en marketing de afilación). 2019. *6 Insider Hacks to Beat Twitter's Algorithm For More Retweets and Followers.* Disponible en: https://www.bigcommerce.com/blog/twitter-hacks/#hack-1-tweet-consistently
43. McGarry, C. (Reportera de tecnología en Tom's Guide). *Twitter bots, fake retweets rake in big bucks.* 2013. Disponible en: https://www.pcworld.com/article/2033766/twitter-bots-fake-retweets-rake-in-big-bucks.html
44. Palacios, F. (Mercadólogo y sociólogo con más de 10 años de experiencia en la docencia universitaria y con más de 25 años de experiencia en marketing turístico). *¿Cuántos tweets al día?* 2017. Disponible en: https://soy.marketing/cuantos-tweets-al-dia/
45. MacCarthy, R. (Científico jefe de datos en KickFactory que dirige operaciones de inteligencia artificial y aprendizaje automático). *The Average Twitter User Now has 707 Followers.* 2016. Disponible en: https://kickfactory.com/blog/average-twitter-followers-updated-2016/

46. Schaffer, N. (Líder reconocido en ayudar a las empresas a maximizar su social como orador principal global, educador universitario, propietario de agencia de redes sociales, autor y consultor de estrategia de redes sociales). *Twitter Followers vs Following: What is the Ideal Ratio?.* 2019. Disponible en: https://nealschaffer.com/twitter-followers-following-quality-or-quantity/
47. Peralta, D. (Social Media Manager, consejero para implementaciones de estrategias de marketing y fundador de la empresa Creative Social Media Solutions). Ratio TFF de una cuenta de Twitter: cómo calcularlo e interpretarlo. 2014. Disponible en: https://davidperalta.es/ratio-tff-de-una-cuenta-de-twitter/
48. Delgado von Eitzen, C. (Ingeniero Superior de Telecomunicación especializado en la creación de estrategias de posicionamiento en internet SEO y analítica web desde 2006). *Perfil del usuario de Twitter en España en 2016 (Resultados).* 2016. Disponible en: http://www.christiandve.com/2016/10/perfil-usuario-twitter-espana-resultados/
49. *Twitter anuncia cuatro nuevas medidas para combatir el spam y la manipulación.* 2018. Moreno, M. (Periodista y fundador de TreceBits, profesor de redes sociales y periodismo, así como colaborador en varios medios de comunicación). Disponible en: https://www.trecebits.com/2018/06/28/twitter-anuncia-cuatro-nuevas-medidas-para-combatir-el-spam-y-la-manipulacion/
50. Documentación oficial de Ubuntu para desarrolladores. Disponible en: https://ubuntu.com/desktop/developers
51. J.Pomeyrol (Persona al frente de la página web MuyLinux). *Ubuntu: qué versión instalar y cuándo actualizar.* Disponible en: https://www.muylinux.com/2018/10/18/ubuntu-version-instalar-cuando-actualizar/
52. Documentación sobre cómo instalar Python 3.5. Disponible en: https://docs.python-guide.org/starting/install3/linux/
53. Documentación oficial sobre PyCharm. Disponible en: https://www.jetbrains.com/pycharm/
54. *PyCharm: uno de los mejores IDE para Python.* Escuela Python. Disponible en: https://www.escuelapython.com/pycharm-uno-de-los-mejores-ide-para-python/
55. Documentación oficial sobre el gestor de paquetes de Python: Pip. Disponible en: https://packaging.python.org/guides/installing-using-linux-tools/#installing-pip-setuptools-wheel-with-linux-package-managers
56. Documentación sobre cómo usar cron y crontab. Disponible en: https://blog.desdelinux.net/cron-crontab-explicados/

57. Documentación oficial sobre la herramienta Pentaho Data Integration (PDI): https://www.hitachivantara.com/es-latam/products/big-data-integration-analytics/pentaho-data-integration.html

Anexo I

A continuación, se procede a listar los 100 usuarios seguidores escogidos para las pruebas del algoritmo que pertenecen a cada partido político. Así pues, los usuarios recolectados para el experimento son los siguientes:

Los 100 seguidores tomados de la cuenta del partido político PSOE:

1. @Miguel98256075
2. @GzVentur
3. @Cguimar81
4. @2perez_laura
5. @Elewis84426176
6. @MentidaLa
7. @AbderrahmaneMg2
8. @bingcelia
9. @DaminJurado5
10. @AhmedAlioui2
11. @JosMara10290523
12. @JoseLuisLuisFe2
13. @diana37352689
14. @Adela22883865
15. @mehdihaririraji
16. @AntonoaPabon
17. @chander78498465
18. @Iris_violin
19. @PaolaRuizMorill
20. @VernaviSoftware
21. @SofiaS86466829
22. @KikoCarrasco4
23. @lutumba_m
24. @Eva20765714
25. @megg_mas
26. @AlbiolManu
27. @jose_fje
28. @JuanCar47572817
29. @peshtenikova
30. @Josehurtadocar4
31. @oscarmargar
32. @PabloCh26303881
33. @ButCatalonia
34. @EstebanApolo1
35. @gatariam
36. @lDavid_Dark
37. @Mariama19721217
38. @lvaro24872110

39. @principepop_92
40. @gregooo_77
41. @ocfba
42. @Zorelor1
43. @Carolvazher
44. @Marcoslc1974
45. @CiudadanosJven1
46. @marininamg
47. @BarryAllaye2
48. @Adnan85432893
49. @schnoerr
50. @BeMa82961180
51. @saravila2003
52. @BelVoces
53. @JoseAntonioSn14
54. @VICTORVhava
55. @jonathanaa_00
56. @VctorDario5
57. @JorobadoH
58. @Mirinda_002
59. @RealMad66359325
60. @FlorTinocoLpez1
61. @RufoBatalla
62. @tabaglez
63. @pabloarmen1
64. @EmilioPaliza1
65. @NAZAKAT17168547
66. @monicaes_1999
67. @Richard71922595
68. @AndresSensei
69. @Elsastrejuridi1
70. @VaniaBreiska
71. @MicasaRepublica
72. @Sallan28812682
73. @DonTemplario
74. @DuquesadeSesa
75. @Francis51058052
76. @Gema10684291
77. @josep_alos
78. @fishecology_
79. @Chohan63367341
80. @Gabriel96146605
81. @AntonioJ1974
82. @igirfer
83. @arielachutegui1
84. @derviche1986
85. @David91164895
86. @Mica75293689

87. @elecciones_y
88. @gabrielatraver
89. @salvado_galvez
90. @albavillegas13
91. @Cirene66142056
92. @juanmarevuelta
93. @lmdcomunismo
94. @Borja_DPe
95. @david_deveze
96. @Irene81759759
97. @lolakjo
98. @bruxa13622341
99. @pipo21230793
100. @ThiernoThierno8

Los 100 seguidores tomados de la cuenta del partido político PP:

1. @Aaronhidalgo111
2. @Nick99P
3. @quiquegf3
4. @JosLuis84186215
5. @Silvia43285954
6. @MANUERGIA
7. @Francis07079679
8. @DolcaValencia
9. @CernudaAlfredo
10. @juanjosegilrey2
11. @MentidaLa
12. @1872Liebknecht
13. @JUANDEFORTUNY
14. @EliassanElias
15. @Rubn50485033
16. @AbderrahmaneMg2
17. @bingcelia
18. @sebasgarnes
19. @AntonoaPabon
20. @Ventura20170402
21. @RaulUron
22. @Iris_violin
23. @estareaquimismo
24. @jasosasass
25. @Elewis84426176
26. @asesinodetroski
27. @jesuslozada9
28. @Eva20765714
29. @megg_mas
30. @AlbiolManu

31. @jose_fje
32. @JuanCar47572817
33. @Arhaan88240349
34. @javiaguilar888
35. @CarmenFages
36. @MataixRequena
37. @corleone6662
38. @gatariam
39. @reliasfernandez
40. @Rudy68548834
41. @OlayaForero
42. @Veronic17282556
43. @IdriisAlpa
44. @hRyg5KGdyrliy2a
45. @JoseAnt88847505
46. @Thania_Escobedo
47. @Joel_Francesqui
48. @jupiter5es
49. @SccGironaProv
50. @CollimF
51. @Tigre_Aguilar_C
52. @nigromante31
53. @Atleti
54. @ClaveriasPablo
55. @AcostaElizabe15
56. @zhandrarengifo1
57. @Tamorlan1
58. @Elchico_Bi
59. @SergioOteo
60. @santiggrosso
61. @RTrebaul
62. @JuanJos98307261
63. @Ginlm
64. @RafaelM29734508
65. @ItFollows3
66. @JoseLui95893704
67. @David91164895
68. @mhrp_ViiiR
69. @colsan05gmailc1
70. @Chino14712901
71. @DavidLM32489496
72. @gduranr1
73. @Zorelor1
74. @Marcoslc1974
75. @Donquij02623167
76. @BarryAllaye2
77. @schnoerr
78. @BeMa82961180

79. @LuciaSalcedoRa1
80. @ero001
81. @7730Lopez
82. @Pomponsito1
83. @PanalGarcia
84. @Evotaaa
85. @Celianistal2
86. @mike_atm
87. @danielaanglada
88. @dresiuqui
89. @PabloDiz6
90. @josesaez955
91. @lailahado
92. @DonTemplario
93. @amartinmalmi
94. @MalangDiedhio20
95. @DavidMu98121019
96. @Gabriel96146605
97. @igirfer
98. @medusa8882
99. @JAperezjurado
100. @YZieza

Los 100 seguidores tomados de la cuenta del partido político Podemos:

1. @marinaconesam1
2. @yanlassua
3. @Guadalu83735256
4. @Ibai_guillen
5. @AlvaroJC_FdeC
6. @Salvado19111829
7. @taxisbarcelona2
8. @CaperucitaVer12
9. @TeoTambriz
10. @Hater96688133
11. @Litus_Maiden
12. @ArribaEsp_Twit
13. @AnnaCarrin1
14. @Aqueronte13
15. @almondat
16. @Ricardo66043511
17. @asumu_obiang
18. @Vincius40123349
19. @DecimMaxim
20. @donramonrius
21. @Pablo50060787
22. @NachoPrezSnche1

23. @manises131273
24. @Mariana16080340
25. @LoubnaZerr
26. @OConciliar
27. @AndairaMir
28. @Miguel_pl01
29. @jaque_in_guada
30. @UxiTreeiBar
31. @diiaz_helena
32. @_davidruizz_
33. @ibizaroig01
34. @IdeologiaT
35. @GIgnacio17
36. @Toni64024795
37. @PattyVelaRo
38. @ARM21438318
39. @Duber_Alvarado
40. @PartidoVerdeCol
41. @kira_45
42. @riotcabrona
43. @luty205
44. @Erick_Layana
45. @19Richard77
46. @CharnecoSFC
47. @prius236
48. @juancadetorres
49. @LUNA36674524
50. @johnyafonso1
51. @DiazGaviola
52. @lublo8
53. @crisazlor
54. @Maika93262525
55. @Manuelyeye1
56. @dobroomir98
57. @Jet_megal
58. @mananamenos
59. @Jose24485461
60. @Chakun16187576
61. @edgar19693
62. @Unfaking_es
63. @anler
64. @chulken390
65. @Unespaol9
66. @_calvo_con_pelo
67. @kemalsyilmaz
68. @MngelsSansFust
69. @antipucherazo
70. @sisqo_b12

71. @mariiiapg4
72. @MiguelRoyo1
73. @juxnespos
74. @ansanguesa
75. @soy_ty
76. @andreatomas
77. @braaavesoul
78. @Alegreluzdelal1
79. @Aniram221
80. @AmparoCasVin
81. @NuriaDescalzo
82. @SebCancio
83. @MRLTR1
84. @Tatiana77812221
85. @JCGE7
86. @JuditSierra3
87. @Evadbfff
88. @DanozJose
89. @Jaume35552482
90. @estapasando__
91. @pedrobenicalap
92. @PeterBe93637394
93. @Lilou89135557
94. @Veronic42311184
95. @morenow2p
96. @AlmenaJose
97. @_m_e_e_l
98. @klausroses
99. @JoseAngelBravo9
100. @Koalilla2

<u>Los 100 seguidores tomados de la cuenta del partido político Vox:</u>

1. @Emilio01377441
2. @72_moustache
3. @GonzaPab
4. @CarmenMisalo
5. @pujol1978
6. @KagawaYusuf
7. @DolcaValencia
8. @PanSarten
9. @novo_rojas
10. @albertoqj74
11. @maneli954
12. @mdrbernal
13. @JuanmaCandela73
14. @BejaranoOtalora

15. @guilaImperial3
16. @RafaVillellas
17. @CarlosP40839310
18. @MAX25297523
19. @Vicen31202584
20. @JotaTol
21. @MissCoc1
22. @JosLuis84186215
23. @almudenaaaaa
24. @MiguelA44240865
25. @JoseGzlez
26. @Letii183
27. @Vicente2692
28. @Acollsamso
29. @Pedro55609651
30. @jlpepegut
31. @juanjosegilrey2
32. @2perez_laura
33. @EliassanElias
34. @Marina84837409
35. @AAlcobeta
36. @AbderrahmaneMg2
37. @montsecordob
38. @79Axterix
39. @bingcelia
40. @fernand04923134
41. @hugo_balbas
42. @vicente43215436
43. @BDragon1986
44. @PIKHOLAN
45. @martinadlaterra
46. @Mamaherny28
47. @rodriigaooo_
48. @julsanchezg
49. @bilisyvinagre
50. @Sergio9Torres
51. @danperexp
52. @AxelrodAriel
53. @sebasgarnes
54. @Mariocasuso1
55. @sevillano021
56. @alfonsom30
57. @Ventura20170402
58. @utmost_catalan
59. @NACHORUIZMIGUEL
60. @alvarovillalon
61. @Angelruiz1979
62. @Damian_OllerH

63. @victorm31250853
64. @jasosasass
65. @Topanga1710
66. @LuzGenil
67. @joss39400
68. @DomanHair
69. @franpollo81
70. @jesuslozada9
71. @mayte_kokoo
72. @Guadalu22065829
73. @David26856664
74. @Trismegistro74
75. @topangalachupa
76. @almagro_vero
77. @MariLlorca6
78. @Elewis84426176
79. @Palmeritas51
80. @papallunaga
81. @Dave80_
82. @tonifoc
83. @jotah50
84. @Diegol_G18
85. @parandoseamirar
86. @Jaime29381510
87. @Antonio75628169
88. @juankarlosok
89. @_patrysm
90. @luisanest
91. @sofimatia
92. @maribelsuarez37
93. @gatariam
94. @elena78894030
95. @CarmenFages
96. @jrbarrerah
97. @jabalimolinero
98. @rnavarro1412
99. @whosthewho
100. @PilarBodega