Universidad de Deusto

# MACRUSS

## Moderación Automática de Comentarios y Reputación de Usuarios en Sitios web Sociales

Jorge de la Peña Sordo

Bilbao, Junio de 2014

UNIVERSIDAD DE DEUSTO

# MACRUSS

## MODERACIÓN AUTOMÁTICA DE COMENTARIOS Y REPUTACIÓN DE USUARIOS EN SITIOS WEB SOCIALES

Tesis doctoral presentada por
JORGE DE LA PEÑA SORDO
dentro del Programa de Doctorado en
INGENIERÍA EN INFORMÁTICA Y TELECOMUNICACIÓN

Dirigida por el
Dr. IGOR SANTOS GRUEIRO
y por el
Dr. PABLO GARCÍA BRINGAS

Autor | Co-director | Co-director

Bilbao, Junio de 2014

*A toda mi familia y a todos mis amigos,*
*gracias por estar siempre ahí,*
*sois los mejores.*

# Resumen

La generación de contenidos por parte de los usuarios supone uno de los mayores valores que la era de la información, tal y como se la conoce hoy en día, aporta a la sociedad. De esta manera, los antiguos portales web, en los que existía un plantel de profesionales que generaba el contenido, se han convertido en plataformas de difusión de la creatividad de los usuarios, los cuales generan contenidos de forma cada vez más profesional. El crecimiento de este tipo de plataformas, organizado en torno a comunidades, es evidente. Prueba de ello es el crecimiento que distintas redes sociales y comunidades han sufrido últimamente.

En particular, diversos sitios web sociales de noticias como *Digg* o *Menéame* son muy populares en este ámbito. Estos sitios trabajan de manera sencilla e intuitiva: los usuarios envían enlaces de artículos en línea, y otros usuarios de este sistema los valoran positiva o negativamente mediante votaciones. Los artículos más votados favorablemente aparecen, finalmente, en la portada.

Esta tesis está enfocada en la web *Menéame*, la cual ya dispone de un método para moderar y así poder filtrar automáticamente comentarios y noticias; sin embargo, está basado en los votos de otros usuarios y, por lo tanto, no puede ser objetivo. Este hecho puede provocar la aparición de grupos de presión y usuarios o grupos trol que desvirtúen el sistema. En la misma línea, hay enfoques para filtrar *spam* en comentarios, donde se propone un método basado en diversas opiniones y características sintácticas para filtrar automáticamente mensajes de *spam* en comentarios, en el sitio web *Amazon*. Para evitar estos problemas, en el contexto de la categorización de usuarios, se han realizado diferentes trabajos empleando las características de las pulsaciones del usuario en el teclado o su actividad en los registros. Sin embargo, no existen trabajos específicos que se encarguen de la problemática de las noticias sociales y sus peculiaridades.

Por ello, esta investigación busca mitigar el salto existente entre la plataforma y el usuario, ya que el volumen de contenido generado en ciertos sitios web no permite que se pueda hacer un control manual. Por lo tanto, proponemos un sistema de moderación automática y escalable que permita a la plataforma tener una mejor información sobre sus usuarios y sobre el uso que hacen de la plataforma. En el mismo sentido, los usuarios estarán más protegidos de agravios y vejaciones, lo que hará que su estancia en la plataforma sea más placentera.

Para su realización hemos diferenciado 4 tareas principalmente. En primer lugar, el estudio de la problemática, en la cual contextualizaremos el problema. En segundo lugar, la categorización de comentarios: (i) entrenamiento y (ii) sistema de categorización. La clasificación de usuarios, en tercer lugar, que haciendo uso de las características de los perfiles de los usuarios y sus comentarios podremos realizar un sistema que los clasifique. Y por último, generaremos un sistema automático para controlar y predecir la reputación y el comportamiento de los usuarios en base a su actividad anterior.

# Abstract

The content generation by users is one of the greatest values that the information age, as it is known today, contributes to society. On this way, the old web sites, in which there was a professional team generating the content, have become platforms for sharing the creativity of users. Increasingly, these users generate available content more professional. The growth of these platforms organized around communities is evident. Recently, the evolution of certain social networks and communities have experienced is a proof of that.

Particularly, social news websites such as *Digg* or its Spanish version *Menéame* are popular social websites. These sites work in a very simple and intuitive way: users submit links to stories online, and other users of those systems rate them by voting their news. Finally, most voted stories appear in the frontpage.

In this dissertation, we focus on *Menéame*. This social news website has already a method for automatically moderate comments and stories in order to filter them. However, it is based on the votes of other users and, therefore, it is not objective since lobbies and trolling collaboration networks may appear distorting the system. In a similar vein, there are approaches to filter spam in reviews. A method based on several opinion and syntactic features to automatically filter spam messages in product reviews was proposed in the website *Amazon*. In the context of user categorization, several works have been done by using the click-features or using activity logs. However, there is no specific work that handles the social news problematic with its particularities.

Therefore, this thesis seeks to mitigate this problem between users and web platforms, due to the huge amount of content data generated in some websites not allows manual control. We propose a new automatic and scalable moderation system allowing the platforms and their administrators to dispose a system that provides

better information about the users and their behaviour. In the same way, such users will also be better protected contemplating offensive content. This fact inspires greater confidence and it will be make a comfortable platform.

Mainly, we have differentiated 4 tasks. Firstly, the study of the issues, in which we contextualize the problem. Secondly, the comments categorization: (i) train and (ii) categorization system. Next, the users' classification using extracted features from the users' profiles and their comments. This fact allows us to implement a system that classifies them. And finally, we generate an automatic system to control and forecast the users' reputation and behaviour based on their previous activity.

# Laburpena

Gaur egun, erabiltzaileek sortutako edukiak, gizartearentzat informazio iturri garrantzitsuenetariko bat suposatzen du. Hori dela eta, aintzinako web portalak non profesionalek edukiak sortzen zituzten, gaur egun eduki profesionalak sortzen daben zabalkunde sormenaren plataformak bihurtu dira. Komunitateen inguruan oinarritutako plataforma hauen hazkundea agerikoa da. Sare sozialak eta komunitateak jaso dituzten hazkuntzak aurrekoan froga dira.

Bereziki, *Digg* eta *Menéame* antzeko albisteen web sozialak oso herrikoiak dira. Web hauek modu errezean eta intuitiboan lan egiten dute: erabiltzaileek *online* dauden artikuluen estekak bidaltzen dituztu eta gero, sistema honen beste erabiltzaileek botuen arabera positiboki edo negatiboki ebaluatzen dituzte. Boto positibo gehiago dituzten artikuluak portadan agertzen dira.

Tesi hau *Menéame* web orrialdean oinarrituta dago. Web honek, istorioak eta komentarioak erabiltzaileen botuen arabera iragazteko eta moderatzeko sistema bat eduki arren, hau ez da objetiboa zeren eta presio taldeek edo *trol* taldeek sistema gutxiesten bait dute. Era berean, komentarioetan *spam*-a iragazteko metodo ezberdinak daude. Adibidez, *Amazon* web orrialdean, iritzietan eta ezaugarri sintaktikoetan oinarritutako metodo bat proposatzen da. Erabiltzaileen mailakapen kontextuan sortzen diren arazo ezberdinak saihesteko, teklatu pulsazioen eta web erregistroetan erabiltzaileak dituen aktibitatearen inguruan lanak egin dira. Hala ere, albiste sozial eta haien arazo bereziek zaintzen dituzte lan espezifikorik ez dago.

Horrela, ikerketa hau web plataforma eta erabiltzailearen arteko tartea estutzen saiatzen da, ezin bait da gaur egun sortzen den informazio guztia eskuz kontrola eramatea. Beraz, erabiltzaile eta haien erabilerei buruzko informazio hobea kudeatzen duen moderazio sistema automatiko eta eskalagarri bat proposatzen dugu. Sistema honen ondorioz, erabiltzaileek kexen eta irainen aurkako tresnan bi-

tartez babestuta egongo dira, web plataforman egondako denbora atseginagoa bihurtuz.

Bere errealizaziorako 4 zeregin desberdin ditugu. Lehendabizi, arazoari buruzko ikerketa. Bigarrenez, komentarioen mailakapena: (i) entrenamendua eta (ii) kategorizazioaren sistema. Hirugarrenez, erabiltzaileen sailkapena, haien ezaugarriak kontutan hartuz mailakatzen duten sistema bat egiten dugu. Eta azkenik, erabiltzaileen ospea eta haien konportamendua kontrolatu eta aurreikusteko sistema automatiko bat sortuko dugu, aurreko aktibitateetan oinarriturik.

# Agradecimientos

Esta sección está reservada para dar las gracias a todas aquellas personas que de una manera u otra han colaborado, ayudado o animado con esta tesis doctoral a lo largo de este periodo de tiempo.

En primer lugar, quiero dar las gracias a Dr. Pablo García Bringas (co-director de esta tesis), por haberme dado la oportunidad en Abril del 2010 de formar parte del laboratorio ahora *DeustoTech Computing* y siempre $S^3Lab$ *(Laboratory for Smartness, Semantics and Security)*, así como de ser partícipe de dicho grupo humano integrado por grandes personas y amigos. También agradecerle por ofrecerme la posibilidad de realizar la estancia predoctoral en Bérgamo, una experiencia única. *Eskerrik asko Pablo.*

En segundo lugar, dar las gracias y un gran abrazo, a mi amigo Dr. Igor Santos Grueiro (co-director de esta tesis). Probablemente, y sin lugar a dudas, sin su ayuda y sin sus ánimos esta tesis doctoral nunca hubiese visto la luz, por ello siempre le estaré muy agradecido. *Eskerrik asko Igor.*

Y qué decir de mis compañeros, de ese fantástico grupo de personas inigualables que forman el $S^3Lab$ y que en su momento me acogieron como un integrante más de esta gran familia: Agustín Z., Aitor S., Álvaro M., Asier G., Borja A., Borja S., Carlos L., Félix B., Guillermo M., Igor R., Iker A., Iker P., Iñigo A., Iván G., Jaime D., Javi N., José G., Juan L., Mikel S., Patxi G., Sendoa R., Sergio H., Xabi C. y Xabi U. *Eskerrik asko denoi.*

No tenía intención de destacar a ningún miembro del equipo, pero veo necesario nombrar a una serie de personas que sin su ayuda esta tesis no hubiese llegado a buen puerto. *Eskerrik asko Iker*, por toda la ayuda prestada y todos los ánimos otorgados. Todavía nos quedan muchos *crawlings* por hacer y muchas *Abadías* por conquistar. *Eskerrik asko Iván*, por tu implicación en esta tesis. Aún nos quedan muchos buenos días y *timbers* que tocar. *Eskerrik asko Asi*, por

ser partícipe de esta tesis, por tu ayuda y por tu paciencia infinita. Algún día vendrás a San Mamés, te lo mereces. *Eskerrik asko Xabi*, por tu colaboración permanente en la elaboración de esta tesis. Ya lo celebraremos con una buena comida, con un filete como no.

También me gustaría agradecer la oportunidad de haber podido acabar de escribir esta tesis en la ciudad italiana de Bérgamo. *Eskerrik asko Giuseppe, grazie mille,* por tu amabilidad y disposición para que durante los 3 meses de estancia predoctoral en el *Dipartimento di Ingegneria dellínformazione e metodi matematici* de la *Università degli studi di Bergamo*, con sede en Dalmine, fuesen perfectos.

Por último, agradecer eternamente a mi familia todo lo que siempre han hecho por mí. *Eskerrik asko aitas*, Álvaro y Ana Mª, por formarme como persona y enseñarme los valores necesarios para vivir esta vida junto a vosotros. *Eskerrik asko a Inés y a mi hermano Jon*, por ser como sois y por los buenos momentos que hemos pasado y pasaremos juntos. *Eskerrik asko aitites y yeyos*, Álvaro y Conchita y José Mª y Angelines, por vuestro apoyo incondicional y confianza plena en mí. *Eskerrik asko familia*, a mis tíos, a mis primos y demás personas que forman parte de esta gran familia, por ser únicos y hacerme feliz continuamente.

Y cómo olvidarme de mis amigos, las mejores personas que he conocido nunca y con las que he pasado, paso y pasaré momentos increíbles en estos años de amistad. A esa cuadrilla memorable de Bilbo (mencionar a D. Arkaitz Tejedo Palacios y D. Xabier Zabalía Jauregi por su ayuda y dedicación en esta tesis), a esa cuadrilla brutal de Villarcayo, a esa cuadrilla impresionante de compañeros en Ingeniería de Telecomunicaciones de la Universidad de Deusto y a ese equipo formidable de balonmano Universidad de Deusto-Loyola Indautxu y a sus *Juniors d´or*, *eskerrik asko lagunok*. Grandes personas y mejores amigos para toda la vida.

Obviamente, alguien me habré dejado en el tintero a la hora de redactar los agradecimientos porque por suerte sois muchos a quienes debo dedicaros esta tesis, porque sois muchos los que pertenecéis a mi vida y me hacéis feliz.

*¡Eskerrik asko denoi!*

# Índice general

# Índice de figuras

# Índice de tablas

*«Internet es mucho más que una tecnología. Es un medio de comunicación, de interacción y de organización social.»*

Manuel Castells Oliván (1942–)

CAPÍTULO

# 1

# Introducción

La Web, tal y como se conoce hoy en día, ha evolucionado a lo largo de los años. Con la aparición del paradigma de la *Web 2.0* (O'Reilly, 2007), la Comunidad de Internet se ha sensibilizado respecto a las principales necesidades del usuario cuando éste navega por la red. Desde entonces, tanto la interacción como la colaboración dinámica por parte de los usuarios ha mejorado drásticamente, comenzando el desarrollo de redes sociales, *wikis* o *blogs*, entre otros. Actualmente, no sólo los administradores de los sitios web son capaces de generar contenidos, sino que también los usuarios del sitio web pueden poner a disposición de otros usuarios un gran volumen de contenidos mostrando sus opiniones o sentimientos acerca de un hecho. Todos los usuarios que deseen publicar diferentes contenidos, lo podrán realizar con rapidez y eficacia. Gracias a esta nueva perspectiva con la que se contempla la red, existe una gran cantidad de información generada por los usuarios, la cual está en constante actualización.

Esta nueva disponibilidad de información puede convertirse en un problema a tener en cuenta; y por ello, la necesidad de filtrar y estructurar los datos mediante nuevos métodos. Consecuentemente, la organización automática de la información y su gestión eficaz se erigen como labores de vital importancia, convirtiéndose en algo indispensable. Todo ello ha causado que cada vez sea más necesario disponer de mecanismos que posibiliten automatizar la clasificación y filtrado de contenidos.

En particular, existen diferentes sitios web sociales de noticias como *Digg*[1] o *Menéame*[2], los cuales son muy populares en este ámbito. Estos sitios trabajan de manera sencilla e intuitiva. En ellos, los usuarios envían enlaces de artículos en línea, y otros usuarios de este sistema los valoran positiva o negativamente mediante votaciones. Finalmente, los artículos más votados favorablemente aparecen en la portada (Lerman, 2007).

Principalmente, esta tesis se enfoca en los sitios web sociales de noticias, tomando como caso de uso el sitio español *Menéame*. Éste, ya dispone de un método para moderar y así poder filtrar automáticamente comentarios e historias. Sin embargo, se basa en los votos de otros usuarios y, por lo tanto, no es objetivo. En esta misma línea, hay diferentes enfoques para filtrar *spam*[3] en comentarios (Jindal y Liu, 2007, 2008), donde los autores proponen un método basado en diversas opiniones y características sintácticas para filtrar automáticamente mensajes de *spam* en comentarios, dentro del sitio web de *Amazon*[4].

El resto del capítulo está estructurado de la siguiente manera. La sección 1.1 explica la motivación por la cual se ha desarrollado esta tesis. La sección 1.2 analiza el concepto de *Web 2.0* y sus aportaciones. La sección 1.3 identifica la problemática por la cual se realiza la investigación. La sección 1.4 establece la hipótesis y los objetivos de esta tesis. La sección 1.5 expone la arquitectura del sistema. La sección 1.6 describe la metodología de investigación empleada. Por último, la sección 1.7 detalla la estructura global del resto de la tesis.

## 1.1 Motivación

La red de redes, más conocida como Internet, se ha convertido en un espacio más sociable en el que todos los usuarios pueden difundir toda clase de aportes y opiniones al resto de internautas mediante el uso de portales web como medio intermediario. En esta percepción de Internet como medio de información, existe una gran variedad no solo de contenidos, sino también de los usuarios que los publican. Cualquier persona de cualquier país, edad o cultura puede expresar con total libertad su opinión acerca de los diversos temas expuestos en la red, pudiendo emplear para ello cualquier clase de información multimedia, no solo textual.

[1] `http://digg.com`
[2] `http://www.meneame.net`
[3] Se denomina *spam* a un mensaje de correo electrónico no deseado o no solicitado.
[4] `http://www.amazon.com`

Como muestra la Figura 1.1, este tipo de información o contenido multimedia está totalmente establecido en la sociedad gracias a la evolución de la tecnología[1]; sin embargo, el contenido textual sigue teniendo un papel de gran relevancia. En este sentido, aunque gran parte de los sitios web sociales dispone de archivos multimedia, las ideas, opiniones y creencias plasmadas por todos los usuarios que acceden al sitio web se encuentran en forma de comentarios. Éstos, pueden ser de diversas clases: siempre hay quien aporta información relevante en su comentario, su propia opinión o ningún género de información; pudiendo ser éste un comentario en tono normal, con un cierto grado de controversia o incluso jocoso.

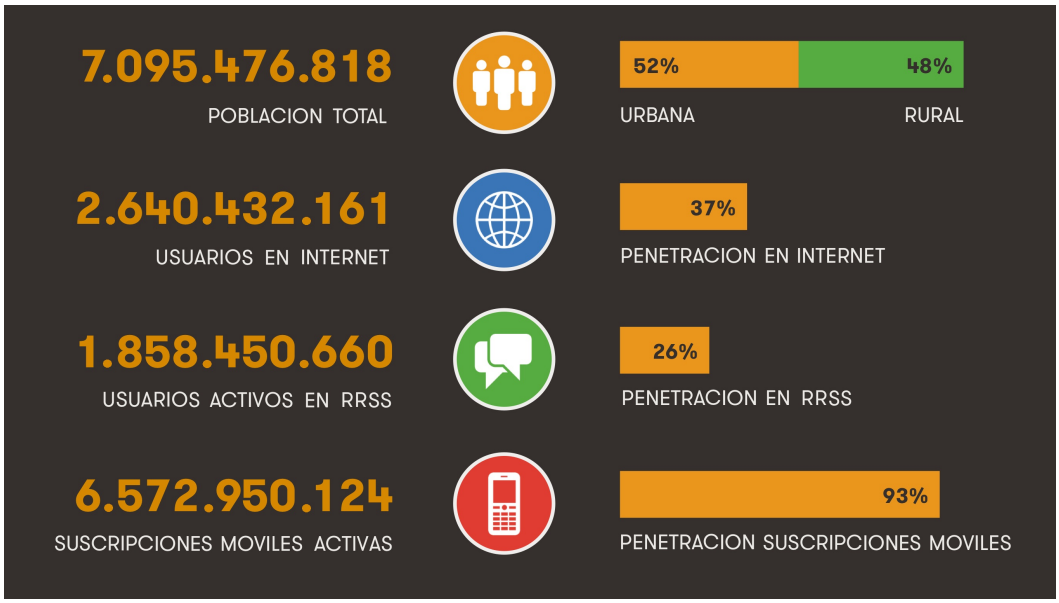



**Figura 1.1:** Adaptación de la gráfica sobre los datos globales de la población mundial en Internet.

Dichos comentarios pueden llegar a ser útiles como base para poder deducir el comportamiento de un usuario. Sin embargo, existen momentos en los que la conducta de las personas puede variar dependiendo de multitud de factores y puntualidades, saliendo de su pauta general. Y al igual que ocurre en el mundo real y en las relaciones sociales, podemos encontrar en la red tanto perfiles de personas similares como totalmente distintas, provocando comportamientos ilícitos y problemas de reputación. A pesar de que Internet es un espacio libre y

[1]Fuente: We Are Social (`http://wearesocial.sg`)

existen términos de libertad de expresión[1], se deben establecer unas pautas que garanticen la integridad de las personas.

Por ello, debido a este tipo de consecuencias, se necesitan herramientas que sean capaces de detectar situaciones extremas y a su vez permitan obtener un control automatizado sobre lo que en todo momento está sucediendo con el contenido y los usuarios en el sitio web. Esto implicaría procesar un gran volumen de datos y sería de gran dificultad realizarlo manualmente. Es por ello que se denota necesario implementar un método capaz de categorizar automáticamente usuarios y sus comentarios, creando un sistema que gestione la reputación de éstos en sitios web sociales.

## 1.2 La *Web 2.0*

El término *Web 2.0* apareció por primera vez en 2005, en una sesión de tormenta de ideas (del término inglés *brainstorming*), por O'Reilly (2005) después de analizar la evolución de Internet tras la explosión de la burbuja tecnológica de las empresas *.com*. En concreto, se definió como:

> «*La Web 2.0 es la red como plataforma, abarcando todos los dispositivos conectados; las aplicaciones Web 2.0 son las que realzan las ventajas intrínsecas de esa plataforma: entregando el software como un servicio continuamente actualizado que mejora cuanto más gente lo utilice, consumiendo y remezclando datos de múltiples fuentes, incluyendo a los propios usuarios individuales, mientras proveen sus propios datos y servicios de manera que permitan ser remezclados por otros, creando redes a través de una "arquitectura de participación", y superando la metáfora de la página propia de la Web 1.0 para ofrecer buenas experiencias por parte de los usuarios.*»

La aparición del término *Web 2.0* hizo posible la definición de sus predecesores: la *Web 1.0* y la *Web 1.5*. Hay que destacar que el cambio ascendente de versiones, no implica la extinción de las tecnologías anteriormente empleadas; sino más bien una serie de mejoras e incorporaciones en las tecnologías existentes, aportando desarrollo e innovación.

[1] `http://www.unesco.org/new/es/communication-and-information/freedom-of-expression/freedom-of-expression-on-the-internet/`

El primero de los conceptos a explicar en este desarrollo temporal de las páginas web, es el denominado *Web 1.0*: son páginas estáticas basadas en el lenguaje de marcado de hipertexto (HTML, acrónimo inglés *HyperText Markup Language*), el lenguaje que interpretan los navegadores web, que no acostumbraban a ser actualizadas asiduamente. Ésta es la Web tradicional que se caracteriza por tener uno o varios administradores web encargados de producir, codificar y publicar, por temas, los contenidos e informaciones del sitio web. Esta información está ubicada en repositorios, donde se debe acudir para acceder a ella, por lo que, son páginas sin interacción directa con el usuario final.

En el año 1997, se puede identificar el cambio hacia lo que hoy conocemos como *Web 1.5*. Durante este periodo, comenzaron a aparecer las primeras páginas web dinámicas, siendo los sistemas de gestión de contenido, gracias a una base de datos actualizada, los encargados de generar las páginas HTML dinámicas creadas *al vuelo* (del inglés *on-the-fly*). A su vez, aparecen lenguajes de programación que permiten obtener datos mediante formularios, siendo éstos los que generan el propio código, permitiendo que cualquier usuario publique contenido en las páginas. Como consecuencia, se empieza a tener en cuenta al usuario, con lo que el aspecto visual tiene mayor importancia en el diseño, con el fin de maximizar el número de visitas de internautas en las páginas web.

Cerca del año 2003, se produce el cambio hacia la *Web 2.0*. Inicialmente, Internet había sido concebido para el trabajo y para las personas con nociones de informática, pero en este instante la red de redes pasa a convertirse en un lugar atractivo para los usuarios, pudiendo intercambiar contenidos libremente. De forma paralela, la generación de contenido se hace más transparente y se permite la distribución y publicación de este contenido en diferentes sitios. También se puede acceder a diversas variedades de contenido multimedia en la red, a la vez que surgen diferentes servicios que permiten difundir contenido, publicándolo de forma instantánea desde diferentes dispositivos (fijos o móviles), con el único requerimiento de tener una conexión a Internet. La nueva Web, es participativa, colaborativa, dinámica y en ella, los usuarios finales se convierten en actores activos, creando, opinando, participando, relacionándose y compartiendo información.

El propio O'Reilly (2007) hizo mención a 7 principios constitutivos de las aplicaciones *Web 2.0*: (i) la Web como plataforma, (ii) el aprovechamiento de la inteligencia colectiva, (iii) los datos son los siguientes *Intel Inside*, (iv) el fin de ciclo de las actualizaciones, (v) el modelo programación ligera, (vi) el software no limitado y (vii) las experiencias enriquecedoras del usuario.

1. **La Web como plataforma.** Se usa un servidor para almacenar la información, así el usuario se conecta a la red en donde tendrá acceso permanente a dicho servidor para poder acceder a los contenidos desde cualquier lugar y compartirlos. La pérdida de información ya no es un problema, ya que la mayor parte de la misma, reside en servidores permanentemente conectados. Uno de los ejemplos más claros de este punto es *Google Apps*[1], donde podemos editar, guardar y compartir documentos, ofreciendo un sistema completo de ofimática en línea.

2. **Aprovechando la inteligencia colectiva.** Cualquier usuario puede aportar conocimiento, mientras que cualquier otro puede corregirlo. Es decir, los usuarios pasan de ser consumidores de información a desarrolladores productivos. La comunidad de usuarios es la encargada de dar forma a los sitios web y a sus servicios, cooperando en base a contenidos. Un ejemplo de este caso es *Delicious*[2], un servicio que permite almacenar los enlaces de distintas páginas web, gestionarlos y compartirlos con otros usuarios. De este modo, los propios usuarios clasifican contenidos web, posibilitando a otros usuarios acceder a ellos gracias a las etiquetas o palabras clave aportadas por la comunidad.

3. **Los datos son los siguientes *Intel Inside*.** En este nuevo contexto, los datos son lo más valioso. El software es abierto o no tan complejo, por lo que obtener un gran grupo de usuarios que comenten o califiquen productos es de gran utilidad e interés, con el fin de aportar información extra al producto, pudiendo aumentar o devaluar el valor comercial del mismo. Las compras por Internet son un problema para la gente desconfiada. Por ello, gracias a este progreso, pueden discernir en sus compras en la medida en que otros usuarios hayan realizado la compra y la hayan comentado positiva o negativamente. En este sentido, *Amazon* es el auténtico referente en la gestión de información de los usuarios como complemento de los productos en venta.

4. **El fin del ciclo de las actualizaciones.** Hoy en día, las empresas *Web 2.0* desarrollan productos software de manera diferente. Antes se lanzaba un producto totalmente acabado, (casi) perfecto y sin ningún tipo de fallos, testeado únicamente por un grupo reducido de clientes. Actualmente, es normal ver cómo las empresas lanzan versiones beta (no finales) de sus

[1] `http://www.google.com/apps`
[2] `http://delicious.com`

productos, esperando las mejoras aportadas por todos los usuarios dispuestos a colaborar con el producto. *WordPress*[1] comenzó siendo un sitio para publicar contenidos, pero gracias a la interacción de un gran número de usuarios, ahora mismo es una plataforma donde se pueden crear sitios más complejos y con un gran número de funcionalidades.

5. **El modelo de programación ligera.** La simplicidad y la eficiencia son los objetivos primordiales. El desarrollo de aplicaciones más pequeñas, más especializadas, adaptables y ligeras ofrecen al usuario un menor tiempo y coste de aprendizaje. Es el caso tanto de *IGoogle*[2] como de *Netvibes*[3], sitios web gratuitos donde podemos personalizar la Web incorporando toda la información o contenido que nos interese: periódicos, *blogs*, el tiempo, correo electrónico, vídeos, fotos, redes sociales, etc.

6. **El software no está limitado a un solo dispositivo.** Desde hace un tiempo, los ordenadores no son los únicos dispositivos donde conectarnos a la red. Existen una multitud de dispositivos electrónicos cotidianos con conexión a Internet (televisores, cámaras fotográficas, videoconsolas, etc.) y como no, los teléfonos móviles y *tablets*, los cuales se han convertido recientemente en los nuevos soportes multimedia con conexión a Internet. El ejemplo más claro en referencia a la conexión móvil es *Twitter*[4]. Podemos escribir comentarios con un máximo de 140 caracteres desde prácticamente cualquier lugar del mundo y adjuntar fotos o indicar la ubicación en la que nos encontramos.

7. **Las experiencias enriquecedoras del usuario.** Antiguamente, se disponía de páginas web estáticas, en las que el texto y las imágenes predominaban. En el presente, nos encontramos con páginas web dinámicas, que gracias al avance de los lenguajes de programación y de la tecnología, mejoran la interactividad de los usuarios con los sitios web mediante contenidos multimedia. El exitoso servicio de almacén de fotografías *Flickr*[5], hizo populares diferentes maneras de interactuar con el contenido publicado: subir imágenes desde el móvil, edición de títulos y descripciones *in situ*, herramienta de organización y creación de álbumes, etc. Y todo ello sin necesidad de recargar la página web.

[1] `http://wordpress.com`
[2] `http://www.google.com/ig`
[3] `http://www.netvibes.com`
[4] `https://twitter.com/`
[5] `http://www.flickr.com`

### 1.2.1 Tipos de aplicaciones *Web 2.0*

Las aplicaciones o servicios a los que ha dado paso el término *Web 2.0* son diversos y aplicables a la mayoría de los ámbitos, para que el usuario sea provisto en todo momento de los contenidos necesarios según su criterio. Como aplicaciones más destacadas hemos encontrado: (i) RSS, (ii) *wikis*, (iii) *blogs*, (iv) redes sociales, (v) fotos, (vi) vídeos y (vii) *folcsonomía*.

- **RSS.** Suscripción Realmente Simple (RSS, del término inglés *Really Simple Syndication* ) es un método para obtener y enviar noticias e información, el cual permite suscribirse a un proveedor de RSS y tener las últimas noticias e información en el propio programa lector de RSS del usuario. Este programa alerta cuando hay nueva información. Este término fue definido por el Consorcio del World Wide Web - W3C[1].

- ***Wikis*.** Es un término hawaiano que significa *rápido*. Una *wiki* es un sitio web colaborativo cuyo contenido puede ser editado por los visitantes del sitio, permitiendo a los usuarios crear y editar fácilmente páginas web colaborativamente (Chao, 2007). El ejemplo más claro es *Wikipedia*[2].

- ***Blogs*.** Proviene de la combinación de las voces inglesas Web y *log*. Es un sitio web que exhibe una serie de funcionalidades distintivas, una dinámica peculiar dominada por una frecuencia de actualización relativamente alta y un formato donde los contenidos se organizan en orden cronológico inverso, apareciendo en primer lugar (arriba) los más recientes (Fumero, 2005). *WordPress*[3] es la herramienta más usada para la creación de *blogs*. Considerando el *microblog* como una manera reducida de *blog*, *Twitter* es una de las aplicaciones con mayor impacto social en este momento.

- **Redes Sociales.** Se definen como un servicio que permite a los usuarios: (i) construir un perfil público o semipúblico dentro de un sistema delimitado, (ii) articular una lista de otros usuarios con los que comparten una conexión y (iii) ver y recorrer su lista de los contactos y las realizadas por otros usuarios dentro del sistema. La naturaleza y nomenclatura de estas conexiones pueden variar de un sitio a otro (Ellison *et al.*, 2007). *Facebook*[4] es en estos momentos la red social más relevante del mundo.

[1] `http://www.w3.org`
[2] `http://es.wikipedia.org`
[3] `http://wordpress.com`
[4] `http://www.facebook.com/`

- **Fotos.** *Flickr*[1] o *Picasa*[2] son aplicaciones que nos permiten organizar, publicar y compartir fotos en línea, así como editar *in situ*.

- **Videos.** Hoy en día, *YouTube*[3] es uno de los sitios más célebre y discutido en Internet y es la primera plataforma que realmente se dirige, de forma masiva, a los usuarios creadores de vídeo (Burgess y Green, 2009).

- ***Folcsonomía.*** Es la unión de las palabras *folc* y taxonomía. *Folc* proviene de *Volks*, palabra alemana que significa *pueblo*. Taxonomía, proviene de los términos griegos *taxis*, cuyo significado es *clasificación*, y *nomos*, que significa *gestionar*. Es un fenómeno emergente de la web social, que surge a partir de los datos referentes a cómo la gente asocia los términos con el contenido que generan, comparten o consumen (Gruber, 2007).

### 1.2.2 El auge de un viejo usuario en la red: el trol

La primera definición del sustantivo trol a la que podemos hacer mención es la citada en la Real Academia Española[4]:

> «*Según la mitología escandinava, monstruo maligno que habita en bosques o grutas.*»

Uno de los primeros usos populares de la palabra trol (del término inglés *troll*), data de la década de 1990, cuando podemos encontrar la palabra trol en la frase *pescando novatos* (de la voz inglesa *trolling for newbies*) en el grupo de *Usenet*, *alt.folklore.urban (A.F.U.)*. Generalmente, es lo que se entiende como una pequeña broma realizada por los veteranos sobre los novatos, que consistía en formular preguntas o proponer temas de conversación tan exagerados que sólo un novato les respondería formalmente.

Este uso de la palabra trol en la década de los 90 sembró un precedente y empezó a usarse el término en Internet a lo largo de los años. Pero con el auge de la red 2.0 y la implicación por parte de los usuarios en la contribución de contenidos, la palabra trol ha tenido una mayor transcendencia. En estos momentos se emplea el término trol en la red 2.0 de diferente modo. Según Turner *et al*.

[1] `http://www.flickr.com`
[2] `http://picasa.google.com`
[3] `http://www.youtube.com/`
[4] `http://www.rae.es`

(2005), un trol es alguien que inicia temas o conversaciones con preguntas aparentemente legítimas, en su mayoría. Sin embargo, el objetivo final de un trol es atraer a otros usuarios, sin saberlo ellos, a discusiones inútiles. Debido a esto, los troles están en riesgo de ser detectados como interrogadores cínicos o manipuladores. En el mismo sentido, Hardaker (2010), el cual analizó discusiones en foros de usuarios, define al trol como la persona que construye una identidad que desea ser parte de un grupo, mientras que en realidad busca como objetivo el interrumpir la propia diversión del grupo. Schwartz (2008), argumenta que los troles son parte de una subcultura de Internet en crecimiento, con una moralidad fluida y un desprecio por casi todo el mundo que se encuentra en línea y afirma que un trol es una persona normal que hace *cosas locas* en Internet. Por su parte Bergstrom (2011), en su investigación sobre los troles en la comunidad de *Reddit*[1], argumenta que la aplicación de la etiqueta trol también puede ser utilizada como una justificación para castigar a quienes transgreden (o son acusados de transgredir) las normas de una comunidad en línea. En la misma línea, en su trabajo sobre detección de cuentas trol en *Twitter* en las elecciones presidenciales de México de 2012, Corona *et al*. (2012) modelan en base a las siguientes características al usuario trol: (i) en general tiene un nombre de usuario poco común, (ii) escribe muchos *tweets*[2] idénticos o hace muchos *retweets*[3], (iii) sigue a pocos usuarios y (iv) es seguido por muy pocas personas, eventualmente, otros troles.

En cuanto a la conducta de este singular tipo de usuarios, Coleman (2010) argumenta que los troles realmente deben ser vistos como algo más parecido a *embusteros*; por lo general, el término trol tiene una connotación negativa dentro de la mayoría de las comunidades en línea. Baker (2001) expresa que el trol puede ser sutil o descaradamente ofensivo con el fin de crear un argumento o como Herring *et al*. (2002) citan, pueden tratar de atraer a otros usuarios haciendo circular una discusión inservible. A causa de esta conducta en la red, Shachaf y Hara (2010) realizaron un estudio en donde exponen las motivaciones o factores que orientan y vigorizan los comportamientos de esta clase de usuarios:

1. El aburrimiento, la búsqueda de atención y la venganza.
2. La diversión y el entretenimiento.
3. El daño a la comunidad y a otras personas.

[1] `http://www.reddit.com`
[2] Un *tweet* es un mensaje de 140 caracteres de *Twitter*.
[3] Un *retweet* es la acción de reenviar un *tweet* de otro usuario.

Debido al surgimiento de los troles y sus comportamientos, se instaura el verbo *trolling* derivado de la acción realizada por dichos usuarios. Cambria *et al*. (2010) explican esta manera de actuar, en el contexto de la web social, como ataques emocionales a una persona o grupo de personas a través de comentarios maliciosos y vulgares, con el fin de provocar su respuesta. Herring *et al*. (2002) denominan esta actuación como un comportamiento que supone atraer a otros a discusiones sin sentido y las cuales requieren mucho tiempo. Donath *et al*. (1999) declaran que es un juego que trata sobre el engaño en la identidad, aunque uno juega sin el consentimiento de la mayor parte de los jugadores; y agrega que puede ser costoso de diferentes maneras: en una comunidad con una tasa alta de engaños, un nuevo usuario puede ser bombardeado con acusaciones airadas y algunos usuarios inocentes nunca podrán volver a participar en dicha comunidad. Para poder acabar con este tipo de comportamiento, Schwartz (2008) añade que sólo se detendrá cuando la audiencia deje de tomar en serio a los troles.

Siguiendo en una tesitura similar, encontramos un término muy ligado al comportamiento de los troles y que muchos usuarios de la red, tanto troles como no, ponen en práctica a menudo: el *flaming* o mensaje hostil, provocador. Lea *et al*. (1992) explican que la noción de que el *comportamiento desinhibido* se asocia con la comunicación a través del ordenador se ha ganado una gran atención. Por su parte, Kiesler *et al*. (1984) sugieren que se produce por la desindividuación o por la disminución de la auto-evaluación: el anonimato de las publicaciones en línea daría lugar a la desinhibición entre las personas. Mientras que otros autores (Lea *et al*., 1992; Postmes *et al*., 1998), mencionan que, si bien el *flaming* y el *trolling* son a menudo desagradables, pueden ser una forma de comportamiento normativo que exprese la identidad social de un grupo de usuarios determinado.

## 1.3 Identificación de la problemática

El aporte de contenido, por parte de los usuarios, constituye uno de los pilares más importantes sobre el cual se sostiene la nueva manera de entender el devenir de la red de redes. La era de la información, tal y como la conocemos hoy en día, está variando su rumbo hacia una perspectiva global más colaborativa, gracias al auge participativo de todos los usuarios en los distintos portales y plataformas de difusión. El aumento de éstas, durante los últimos años, es más que notorio. Como ejemplo, cabe destacar la proliferación tanto de redes sociales como de comunidades o foros de intercambio de opiniones. En la Figura 1.2,

se puede observar como el crecimiento de las redes sociales en un periodo de 5 años[1], prácticamente duplicó al crecimiento del propio Internet. Lo cual nos hace deducir que pronto toda persona con conexión a Internet, tendrá en su haber un perfil en cualquier red social.

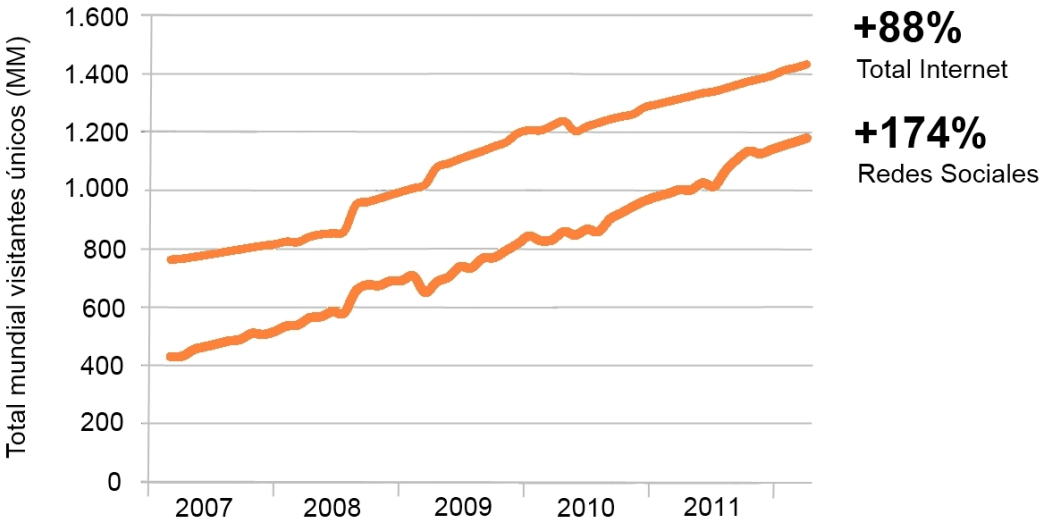



**Figura 1.2:** Adaptación de la gráfica sobre el crecimiento de la audiencia mundial de redes sociales.

Sin embargo, la progresión de esta clase de comunidades provoca un cambio de rol en cuanto a quién y cómo se genera el contenido. Mientras antaño los encargados de controlar y originar la información a mostrar en la red era un equipo de profesionales dedicados a ello, en estos momentos son los usuarios los que generalmente se ocupan de esta tarea y cada vez de una manera más experta y creativa.

Y esta variación en quién es el encargado de la creación de contenidos, tiene una serie de aspectos relevantes. Ya que el volumen de contenido aumenta de manera exponencial, el nivel de exigencia al que están expuestas estas plataformas es mayor. Por ello, ciertas características técnicas, como el poder atender en tiempo real toda clase de peticiones desde cualquier dispositivo conectado o el gran incremento de tráfico debido a dichas peticiones, deben ser más flexibles y menos autoritarias.

Y he ahí uno de los problemas actuales: la enorme ola de información que habilita a ciertos usuarios a difundir contenido polémico, hiriente o incluso delictivo. Mientras que hace unos años esta consecuencia sería debatida entre los

[1]Fuente: comScore Media Metrix

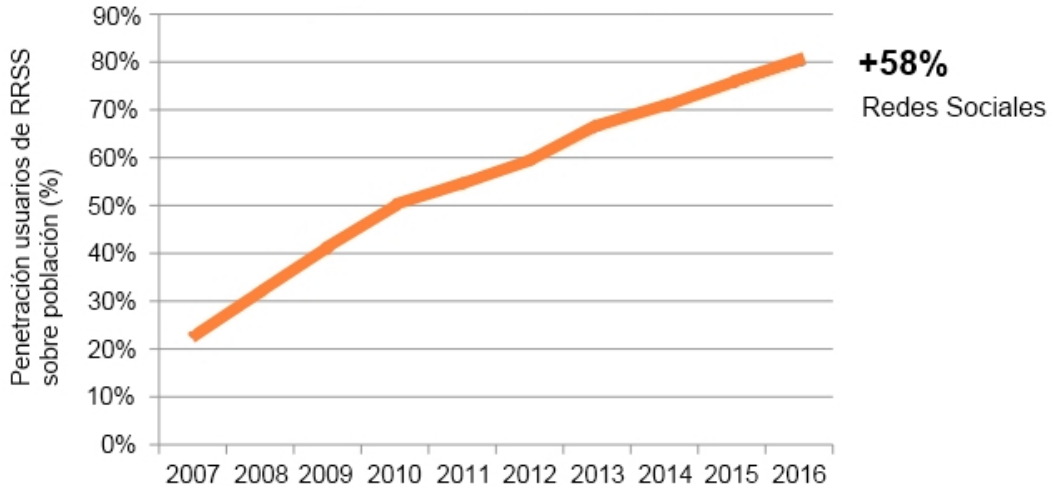



**Figura 1.3:** Adaptación de la gráfica sobre la penetración de usuarios en redes sociales en España.

administradores del sitio o plataforma web para su eliminación, previa a la publicación, en estos momentos se presenta inasequible el poder controlar y discernir en todo momento y en tiempo real mediante un grupo de trabajo humano, si un contenido es malicioso o no. Se necesitaría un importante grupo de trabajo, en cuanto a número de integrantes, que debería de estar permanentemente alerta para juzgar si una publicación es legítima o no, la posterior responsabilidad de sus actos y el gran gasto que todo ello conllevaría para las empresas. Y este grupo de trabajo debería de contar cada vez más con un mayor número de participantes, ya que como muestra la Figura 1.3, se puede observar un gran crecimiento: desde el 22 % de usuarios perteneciente al 2007 hasta el 80 % previsto para 2016[1].

Por ello, esta investigación busca paliar este problema entre los distintos usuarios y las plataformas web, proponiendo un nuevo sistema de moderación automática y escalable que permita a las plataformas y a sus gestores disponer de una herramienta que aporte mayor información sobre dichos usuarios y su comportamiento. Y por otra parte, los propios usuarios también se verán más protegidos de contemplar contenido ofensivo, lo que inspirará mayor confianza y confortabilidad a la hora de navegar por el sitio web.

[1]Fuente: Strategy Analytics

## 1.4 Hipótesis y objetivos

Una vez identificada la problemática en cuestión, hemos constituido la hipótesis fundamental de partida sobre la cual se basa esta tesis:

> «*Es posible determinar, gestionar y predecir el comportamiento de los usuarios en diferentes plataformas web mediante el empleo de minería de opiniones, técnicas de aprendizaje automático, métodos de recuperación de información y series temporales.*»

Con la hipótesis de partida planteada, hemos procedido a identificar el objetivo principal de esta investigación:

**Objetivo principal 1.** *Desarrollar un sistema gestor y predictor de reputaciones de usuarios en sitios web sociales mediante la categorización automática de usuarios y sus comentarios.*

Este objetivo principal, a su vez, puede ser dividido en diferentes objetivos específicos, con el propósito de cumplir tal objetivo principal:

**Objetivo específico 1.** *Desarrollar un método para categorizar comentarios publicados por usuarios en sitios web sociales.*

**Objetivo específico 2.** *Diseñar y desarrollar nuevas técnicas para la clasificación y seguimiento de usuarios.*

**Objetivo específico 3.** *Implementar métodos eficaces para la gestión de la reputación de los usuarios y predicción de su comportamiento.*

El primer objetivo incluye todas aquellas acciones que el sistema necesita para la categorización de comentarios, incluyendo el entrenamiento del módulo y un sistema de categorización. En cuanto al segundo objetivo, utilizando las características de los perfiles de los usuarios y en función de los comentarios que éstos publican, se realizará un sistema que clasifique a los usuarios en alguna de las categorías establecidas previamente. Finalmente, en el tercer objetivo específico, se generará un sistema para la gestión de la reputación de los diferentes usuarios, el cual pueda predecir comportamientos futuros de los mismos.

Con la meta de cumplimentar estos objetivos específicos, los hemos desglosado en ciertos objetivos operacionales. A continuación se enumeran:

**Objetivo Operacional 1.** *Desarrollar un sistema para la categorización automática de comentarios en sitios web sociales.*

**Objetivo Operacional 2.** *Establecer las diferentes categorías para este sistema.*

**Objetivo Operacional 3.** *Implementar un nuevo método de categorización de usuarios.*

**Objetivo Operacional 4.** *Desarrollar una nueva técnica para el seguimiento de los usuarios.*

**Objetivo Operacional 5.** *Adecuación de los métodos de series temporales para la predicción temporal.*

**Objetivo Operacional 6.** *Realización de una prueba de concepto sobre predicción de acciones futuras de usuarios.*

**Objetivo Operacional 7.** *Generación de un modelo de reputación para los usuarios.*

Estos objetivos operacionales son las pautas a seguir durante la investigación. El cumplimiento de todos ellos, supondrá la demostración de la hipótesis inicial anteriormente citada.

## 1.5 Arquitectura general de la solución

En esta sección se presenta la arquitectura del sistema propuesto, para el cumplimiento de los objetivos y la hipótesis anteriormente establecida. Como muestra la Figura 1.4, la arquitectura consta de una serie de módulos, explicados a continuación:

- **Clasificación de comentarios.** Este módulo de categorización de comentarios utiliza un modelo de aprendizaje automático, entrenado previamente con un conjunto de datos formado por comentarios. Para ello, el clasificador categorizará en base a las clases propuestas: (i) el tipo de comentario, (ii) el enfoque del comentario y (iii) el grado del comentario.
- **Clasificación de usuarios.** Al igual que en el caso anterior, este clasificador categorizará los diferentes usuarios entrantes mediante el empleo de un modelo de aprendizaje automático. En este caso, se entrena con un

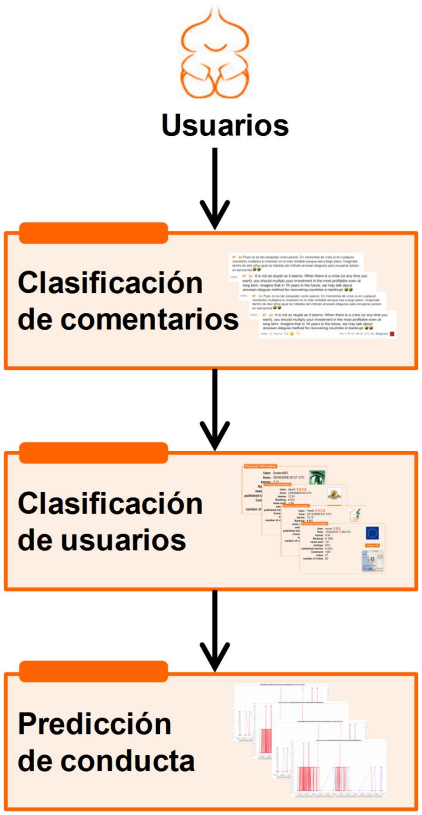



**Figura 1.4:** Arquitectura del sistema sugerido.

conjunto de datos formado por la información del perfil del usuario y sus comentarios previamente clasificados en la etapa anterior. La clases que nos definirán a un usuarios son las siguientes: (i) el tipo de usuario y (ii) el grado del usuario.

- **Predicción de la conducta.** Como último módulo del sistema se encuentra el predictor de conducta. Este componente predirá el comportamiento futuro de un usuario, pronosticando los próximos *n* comentarios del mismo y su grado, gracias a la clasificación previamente realizada en las etapas anteriores.

El sistema actúa de manera secuencial; es decir, las tareas se ejecutan una detrás de otra y hasta que una de ellas no haya finalizado, la siguiente no podrá comenzar. Esto se debe principalmente, a que en los módulos: (i) de categorización de usuarios y (ii) de predicción de la conducta, la entrada a un componente es la salida del precedente.

En primer lugar, se selecciona un usuario y sus comentarios como entrada del sistema, para proceder a determinar si éste es un usuario conflictivo o no. Para ello, y como primer paso, el módulo categorizador de comentarios clasificará los comentarios que el usuario haya publicado hasta la fecha, en base a las clases definidas. Con ello, podremos discernir los tipos de comentario que redacta el usuario para posteriormente emplearlo como entrada del siguiente módulo.

En segundo lugar, el componente de categorización de usuarios clasificará al usuario en base a sus comentarios y a su actividad en el sitio web, registrada en su perfil, para poder llegar a saber si el comportamiento general de un usuario es polémico o no y en consecuencia, si debe ser restringido a causa del mismo.

En último lugar, el módulo de predicción del comportamiento, pronosticará la conducta de dicho usuario en base a su historial de comentarios y las características de su perfil, anteriormente clasificadas. Nos devolverá el tipo de comentario que realizará y la fecha en la cual lo publicará. De esta manera, podremos decidir bloquear a un usuario sabiendo tanto su comportamiento anterior como posterior.

## 1.6 Metodología de la investigación

La metodología con la que pretendemos llegar a completar los objetivos de esta investigación doctoral, debe de estar sustentada en diversos pilares:

- La **reproducibilidad**, es decir, la capacidad de poder repetir un experimento determinado o varios, por cualquier otra persona o grupo de personas de manera que dichos experimentos obtengan resultados lo más similares posibles al realizado por el experto. Este fundamento estará basado, principalmente, en la comunicación y publicidad de los resultados logrados.

- La **falsabilidad** o **refutabilidad**, es otro de los ejes con mayor relevancia a la hora de ejecutar la investigación doctoral. Este fundamento se refiere a que toda proposición científica ha de ser susceptible de ser rebatida por otros científicos mediante un contraejemplo o una observación empírica. Esto implica que se pueden diseñar experimentos que en el caso de ofrecer resultados adversos a los predichos, negarían la hipótesis lanzada.

En este caso, el método de investigación que aplicaremos para validar dicha hipótesis inicial, será cuantitativo. Debido especialmente a que las evidencias

que se consiguen pueden ser medibles. Para finalizar, exponemos los pasos necesarios para la realización de la tesis doctoral:

1. **Identificar el problema y su posible solución.** En este primer paso de nuestra metodología de investigación, contextualizaremos el problema de una manera precisa y exacta, realizando los estudios pertinentes de las técnicas a adaptar.

2. **Adquirir el conocimiento necesario.** Para ello deberemos de obtener una base de conocimientos globalizada acerca del tema sobre el que se va a profundizar la investigación. Una vez obtenido dicho entendimiento global, avanzaremos hasta un conocimiento más específico y concreto dentro del área de investigación en la cual nos encontramos.

3. **Dividir los retos a superar.** En este punto de nuestra metodología, dividiremos los desafíos que nos plantea la problemática a afrontar, con el fin de minimizar el impacto de la misma. En consecuencia, y gracias a este método, podremos ocuparnos de problemáticas de menor complejidad y dificultad con el avance que ello supone. Cabe destacar que la realización de estas tareas divididas, deberá de implementarse de manera secuencial, ya que el hecho de ejecutarlas paralelamente podría originar la pérdida de información entre estos retos; con lo cual, quedaría en entredicho el uso de esta técnica y la posterior solución de la problemática definida. A continuación, enumeramos las diferentes etapas de esta división:

   (a) *Adquisición de conocimiento más específico*. Realizaremos un profundo estudio de los temas en cuestión, una vez declarados y expuestos los diversos problemas.

   (b) *Definición del experimento*. En esta etapa, haremos una definición sobre la investigación que vamos a realizar, así como de los procedimientos a emplear. Como consecuencia, lograremos conseguir la instrumentación necesaria para el desarrollo del experimento y la posterior medición de sus resultados.

   (c) *Evaluación de resultados intermedios*. Adquiriremos los resultados logrados, gracias a la experimentación efectuada en este paso de la metodología.

   (d) *Análisis de los resultados obtenidos*. Finalmente, y con los datos logrados en la etapa anterior, procederemos a analizarlos. Para ello,

emplearemos la instrumentación adquirida anteriormente, para poder solucionar el problema definido mediante una evaluación de los resultados.

4. **Analizar e interpretar los resultados totales.** En esta etapa de nuestra metodología, agruparemos e interpretaremos los resultados obtenidos para poder constatar que las etapas previas han sido solucionadas con éxito. Con ello, buscaremos solucionar el problema inicialmente planteado en nuestra metodología de investigación.

5. **Difundir los datos resultantes.** En último lugar, y para finalizar este proceso metodológico de nuestra tesis, comenzaremos la divulgación de las conclusiones de la experimentación dentro de la comunidad científica, compartiendo el conocimiento adquirido durante esta investigación.

## 1.7 Estructura de la tesis

Esta sección presenta los capítulos restantes en los que la tesis doctoral *Moderación Automática de Comentarios y Reputación de Usuarios en Sitios web Sociales* está dividida:

- **Capítulo 2. Estudio de la literatura.** Este capítulo expone, de manera resumida, el estado del arte en el que se encuentran las diferentes técnicas empleadas en la investigación. Diversos métodos de procesado de datos y textos, así como distintos métodos de filtrado de contenidos y textos y algunos de los métodos de reputación y relevancia más notables son desarrollados a lo largo de este capítulo.

- **Capítulo 3. Modelado y procesado inicial de los datos.** Esta sección de la tesis doctoral introduce al sitio web social de noticias elegido: *Meneame.net*. A continuación, la sección explica el procesado inicial de los datos mediante la adquisición de los mismos y la descripción de los conjuntos de datos (comentarios y usuarios). Para posteriormente, comenzar el proceso de etiquetado de los conjuntos de datos. Una vez etiquetados los datos correctamente, la adaptación de éstos mediante la extracción de ciertas características, tanto de los comentarios como de los usuarios, es necesaria para la siguiente etapa.

- **Capítulo 4. Categorización automática de comentarios y usuarios.** Este cuarto apartado de la investigación, en un primer momento describe

los algoritmos de aprendizaje supervisado utilizados a lo largo de la experimentación realizada. Más adelante, expone la validación empírica y la metodología llevado a cabo con el fin de evaluar los resultados obtenidos tanto en el conjunto de comentarios como en el conjunto de usuarios.

- **Capítulo 5. Reducción del esfuerzo de etiquetado.** Los algoritmos semi-supervisados, y más concretamente la clasificación colectiva, son evaluados en este capítulo con el fin de minimizar los esfuerzos de etiquetado para ambos conjuntos de datos (comentarios y usuarios), de tal manera que los resultados logrados con anterioridad no disminuyan su porcentaje de acierto.

- **Capítulo 6. Análisis y predicción de series temporales.** Esta sección detalla los pasos a seguir para poder pronosticar las series temporales elaboradas sobre ciertos usuarios de *Menéame*, una vez categorizados tanto los propios usuarios como sus comentarios. Dichas series temporales serán evaluadas en base a la tasa de error absoluto medio y la tasa de la raíz del error cuadrático medio, después de haber sido realizada la correspondiente experimentación mediante los modelos predictivos de series temporales seleccionados.

- **Capítulo 7. Conclusiones.** El capítulo final de la tesis presentada establece las principales contribuciones realizadas repasando de nuevo la hipótesis fundamental y los correspondientes objetivos. A su vez, también plantea las limitaciones de nuestro trabajo, así como las futuras líneas de investigación y las consideraciones finales.

*«El mayor enemigo del conocimiento no es la ignorancia, sino la ilusión del conocimiento.»*

Stephen William Hawking (1942–)

CAPÍTULO

# 2

# Estudio de la literatura

A lo largo de los últimos años, el rápido desarrollo que ha experimentado Internet ha facilitado el acceso a toda clase de contenidos y por lo tanto, a un mayor número de fuentes de información. Este hecho implica que el volumen disponible en Internet y en *intranets*[1] corporativas continúa creciendo y por consiguiente, existe la necesidad de disponer de herramientas de ayuda para los usuarios; con el fin de encontrar, filtrar y gestionar mejor los recursos. Consecuentemente, la estructuración automática de la información desempeña un papel de vital importancia y la gestión eficiente se ha convertido en algo imprescindible.

En particular, se ha producido un crecimiento masivo en el volumen de datos, especialmente en los datos de texto, en diferentes aplicaciones web, como los medios de comunicación social. Mientras que las aplicaciones clásicas se han centrado en el procesamiento y minería de texto en bruto. La llegada de estas aplicaciones web requiere nuevos métodos para la minería y el procesamiento de texto, tales como la información plurilingüe o la conjunción de la minería de texto con otros tipos de datos multimedia, como las imágenes o los vídeos.

En el mismo sentido, el rápido crecimiento de las redes de comunicación, aparte del aumento del contenido disponible en la red, también ha estimulado el desarrollo de numerosas aplicaciones colaborativas. Debido a ello, la repu-

[1] Una intranet es una red local privada o interna de ordenadores que emplean protocolos y software de Internet para compartir datos (empleado en empresas, organizaciones, etc).

tación y la confianza de los usuarios desempeñan un papel fundamental en tales aplicaciones, al permitir a múltiples partes establecer relaciones para lograr alcanzar un beneficio mutuo. En general, la reputación es la opinión del público hacia una persona, un grupo de personas o una organización. En el contexto de las aplicaciones colaborativas, la reputación permite a las partes crear cierta confianza entre ellas; es decir, el grado de confianza que una parte tiene sobre otra con un propósito o decisión determinada.

Por estas razones, se nos antoja necesario el estudio de diferentes técnicas y procedimientos, como los motores computacionales de reputación o ciertos sistemas de reputación comerciales, para el correcto tratamiento y gestión de los diferentes tipos de usuarios que perviven en la red, así como su comportamiento impredecible frente a cualquier tipo de acontecimiento o incidente que haya sucedido, esté sucediendo o sucederá en los distintos sitios web sociales.

El resto del capítulo está estructurado de la siguiente manera. La sección 2.1 revisa a los métodos comúnmente utilizados para el procesado de datos y textos. La sección 2.2 desarrolla algunas de las técnicas de filtrado de contenidos y textos. La sección 2.3 identifica una serie de métodos para evaluar la reputación y relevancia de los usuarios en sitios web. La sección 2.4 instaura el concepto de series temporales y la predicción. Por último, la sección 2.5 resume los conceptos expresados a lo largo de este capítulo.

## 2.1 Métodos de procesado de datos y textos

### 2.1.1 El descubrimiento de conocimiento

El descubrimiento de conocimiento (KD, de las voces inglesas *Knowledge Discovery*), o descubrimiento de conocimiento en bases de datos (KDD, de los términos ingleses *Knowledge Discovery Databases*) se define según Fayyad *et al*. (1996) como:

> «*El descubrimiento de conocimiento en bases de datos, es el proceso no trivial de identificación de patrones válidos, innovadores, potencialmente útiles y en último lugar, comprensibles, en los datos.*»

El descubrimiento de conocimiento en bases de datos es un proceso que se define mediante varios pasos que tienen que ser aplicados a un conjunto de datos de interés, para extraer patrones útiles. Estos pasos tienen que ser

realizados de manera iterativa y por lo general, varios de éstos requieren una respuesta interactiva por parte del usuario. El análisis de datos en KDD pretende encontrar patrones ocultos y conexiones en esos datos. Las características que pueden ser usadas para medir la calidad de los patrones encontrados en los datos son: (i) la inteligibilidad para las personas, (ii) la validez en el contexto de las medidas estadísticas dadas, (iii) la novedad y (iv) la utilidad.

### 2.1.2 La minería de datos

La minería de datos es un campo en el que se han observado rápidos avances en los últimos años (Han y Kamber, 2006). Estos avances son debido a los grandes adelantos realizados en la tecnología *software* y *hardware*, los cuales han hecho posible la disponibilidad de diferentes tipos de datos. La investigación en el campo del descubrimiento de conocimiento y minería de datos se encuentra todavía en una situación de cambio constante, ya que a menudo se utilizan los términos de manera confusa. Por un lado, está la minería de datos como sinónimo de KDD, lo cual significa que la minería de datos contiene todos los aspectos del proceso de descubrimiento de conocimiento. Por otra parte, se considera la minería de datos como parte del proceso del descubrimiento de conocimiento (Fayyad *et al*., 1996) y a su vez describe la fase de modelado.

Las raíces de la minería de datos se extienden a través de diversas áreas de investigación que destacan el carácter interdisciplinario de este campo. A continuación, hemos resumido brevemente las relaciones con algunas de las áreas de esta investigación: (i) las bases de datos, (ii) el aprendizaje automático y (iii) la estadística.

- **Bases de datos.** Son necesarias para gestionar eficientemente grandes cantidades de datos. Una base de datos representa no solo el medio para almacenar y acceder consistentemente a los datos, sino el poder soportar análisis de datos con algoritmos de minería de datos. Por lo tanto, el uso de esta tecnología en los procesos de minería de datos puede ser útil. Una visión global de la minería de datos desde la perspectiva de base de datos, la hemos observado en Chen *et al*. (1996).

- **Aprendizaje automático.** El aprendizaje automático (ML, del inglés *Machine Learning*) es un área de la inteligencia artificial (AI, de las voces inglesas *Artificial Intelligence*), que trata sobre el desarrollo de técnicas que permiten a los ordenadores *aprender* mediante el análisis de conjuntos de

datos. El enfoque de la mayoría de los métodos de aprendizaje automático se encuentra dirigido a los datos simbólicos. El aprendizaje automático también se ocupa de la complejidad algorítmica en implementaciones computacionales. Mitchell (1997) presenta varios de los métodos comúnmente usados en ML.

- **Estadística.** Tiene su fundamento en las matemáticas y se ocupa de la ciencia y la práctica en el análisis de los datos empíricos. Está basada en la teoría estadística, que es una rama de las matemáticas aplicadas. Dentro de la teoría estadística, la aleatoriedad y la incertidumbre son modeladas por la teoría de probabilidades. Hoy en día, muchos métodos estadísticos se utilizan en el campo de KDD. Se pueden observar resúmenes en (Friedman *et al.*, 2001; Berthold y Hand, 2000; Maitra, 2002).

### 2.1.3 La minería de textos

La Web es un agente *habilitador*, tecnológicamente hablando, que fomenta la creación de una gran cantidad de contenido textual por diferentes usuarios, gracias a la facilidad y sencillez con la que se guarda y procesa este contenido textual. La creciente cantidad de datos de textos disponible en diversas aplicaciones ha creado la necesidad de realizar avances en el diseño de algoritmos que sean capaces de aprender patrones de los datos de una manera dinámica y escalable.

Mientras los datos estructurados son manejados por sistemas de bases de datos, generalmente, los datos de textos son gestionados a través de un motor de búsqueda, debido a la falta de estructuras (Croft *et al.*, 2010). Un motor de búsqueda permite al usuario encontrar información útil en una colección con una consulta de palabras clave. Cómo mejorar la efectividad y eficiencia del propio motor ha sido uno de los temas centrales a investigar en el campo de la recuperación de información (Baeza-Yates *et al.*, 1999; Jones, 1997).

Sin embargo, tradicionalmente la investigación sobre recuperación de la información se ha centrado más en facilitar el acceso a la información (Jones, 1997) en lugar de analizar la información para descubrir patrones, los cuales son el objetivo principal de la minería de textos.

La minería de textos o descubrimiento de conocimiento a partir de texto (KDT, de los términos ingleses *Knowledge Discovery from Text*), mencionado por primera vez por Feldman y Dagan (1995), consiste en el análisis automático de textos empleando técnicas de:

- Recuperación de información (IR, del inglés *Information Retrieval*).
- Extracción de información (IE, de las voces inglesas *Information Extraction*).
- Procesado de lenguaje natural (NLP, de los términos ingleses *Natural Language Processing*).

Estas técnicas son explicadas posteriormente en la sección 2.1.4. A su vez, la minería de texto también tiene conexión con los algoritmos y métodos anteriormente expresados en las secciones 2.1.1 y 2.1.2 :

- El descubrimiento de conocimiento en bases de datos.
- La minería de datos.
- El aprendizaje automático.
- La estadística.

Un aspecto importante de la minería de textos es su continua actualización e innovación, ya que existen diversos grupos de investigadores que trabajan en los nuevos aspectos de la minería de textos, en el contexto de plataformas emergentes como los flujos de datos y las redes sociales.

### 2.1.4 Otras áreas de investigación relacionadas

La selección de características, la influencia del conocimiento del dominio y los procedimientos específicos del dominio desempeñan un papel importante en la minería de textos. Por lo tanto, es necesario una adaptación de los algoritmos de minería de datos conocidos a datos de texto basándose en la experiencia y los resultados de la investigación en la recuperación de información, procesamiento del lenguaje natural y la extracción de información. En todas estas áreas, también se aplican métodos de minería de datos y estadísticas para gestionar sus tareas específicas.

#### 2.1.4.1 La recuperación de información

La recuperación de información (IR, del inglés *Information Retrieval*) es la obtención de documentos que contienen las respuestas a las preguntas realizadas y

no la obtención de respuestas por sí mismas (Hearst, 1999). Con el fin de lograr este objetivo, se utilizan medidas y métodos estadísticos para el procesamiento automático de los datos de texto y la comparación con la pregunta realizada. La recuperación de información, en un sentido más amplio, se ocupa de toda la gama de procesamiento de información, desde la recuperación de datos hasta la recuperación de conocimiento (Jones, 1997). A pesar de que la recuperación de información es una área de investigación relativamente antigua, donde los primeros intentos para la indexación automática se hicieron en 1975 (Salton *et al.*, 1975), logró una mayor atención con el creciente desarrollo de la red informática mundial (WWW, de la voz inglesa *World Wide Web*) y la necesidad de motores de búsqueda más sofisticados.

Aunque la definición de recuperación de la información se basa en la idea de preguntas y respuestas, los sistemas que recuperan documentos basados en palabras clave, es decir, los sistemas que llevan a cabo la *recuperación de documentos* como la mayoría de los motores de búsqueda, son frecuentemente denominados sistemas de recuperación de información.

#### 2.1.4.2 **El procesamiento del lenguaje natural**

El procesamiento del lenguaje natural (NLP, de las voces inglesas *Natural Language Processing*), estudia los problemas relacionados con el procesamiento y manipulación de los lenguajes naturales. El NLP pretende obtener el conocimiento acerca de cómo las personas entienden y usan el lenguaje, con el fin de implementar herramientas y técnicas para conseguir que las aplicaciones informáticas puedan entenderlo y gestionarlo.

Resumidamente, el objetivo general del procesado de lenguaje natural es lograr una mejor comprensión del propio lenguaje natural mediante el uso de ordenadores (Kodratoff, 1999). Otros autores incluyen también el empleo de técnicas sencillas y duraderas para el procesado rápido de texto, tal como presentan Berwick *et al.* (1991). Sus bases habitan en una amplia serie de disciplinas: ciencias de la información, matemáticas, lingüística, inteligencia artificial, robótica, psicología, etc.

El rango de técnicas utilizadas se extiende desde la simple manipulación de cadenas de texto hasta el tratamiento automatizado de consultas en lenguaje natural. Además, las técnicas de análisis lingüístico se utilizan, entre otras cosas, para el procesamiento de texto en lenguaje natural, interfaces de usuario y reconocimiento de voz, entre otras.

#### 2.1.4.3 **La extracción de información**

El texto en el lenguaje natural contiene mucha información que no es adecuada para el análisis automático por parte de un ordenador. Sin embargo, los ordenadores pueden ser usados para filtrar grandes cantidades de texto y extraer la información útil a partir de palabras aisladas, frases o párrafos. Por lo tanto, la extracción de información (IE, de los términos ingleses *Information Extraction*) puede ser considerada como una forma restringida dentro de todo el lenguaje natural, donde sabemos de antemano qué tipo de información semántica estamos buscando. La tarea principal es extraer parte del texto y asignar atributos específicos al mismo; mientras que el objetivo principal de los métodos de IE es la extracción de información específica de documentos de texto. Éstos se almacenan en bases de datos con patrones similares (Wilks, 1997) y por lo tanto, están disponibles para su posterior uso.

La tarea de extracción de información se descompone en una serie de etapas de procesamiento, que incluyen:

- La *tokenización*[1].
- La segmentación de la oración.
- La asignación de *parte de la oración* (POS, de las voces inglesas *Part Of Speech*)[2].
- La identificación de entidades específicas; es decir, nombres de personas, nombres de lugares y nombres de organizaciones.

Algunas frases y oraciones de mayor nivel tienen que ser analizadas, semánticamente interpretadas e integradas. Aunque los sistemas de información más precisos de extracción a menudo implican módulos manuales de procesado de lenguaje, se han logrado progresos en la aplicación de técnicas de minería de datos a este proceso.

[1] *Tokenizar* es la acción de dividir un flujo de texto en palabras, frases, símbolos u otros elementos denominados *tokens*.

[2] El *Part Of Speech* es la categorización de las palabras según su comportamiento sintáctico o morfológico.

## 2.2 Métodos de filtrado de contenidos y textos

### 2.2.1 El filtrado de contenidos

La necesidad de filtrar información con el fin de proteger a los usuarios frente a posibles contenidos nocivos, puede ser considerada como uno de los problemas sociales más urgentes. Estos problemas vienen derivados de la transformación de la Web en un espacio de información pública. A pesar de que la clasificación de la Web y los sistemas de filtrado se han desarrollado y puesto a disposición del público desde muy temprano, no se ha establecido hasta ahora un método eficaz debido a la insuficiencia de soluciones propuestas. El filtrado web es entonces un área de difícil investigación que necesita la definición y aplicación de nuevas estrategias, teniendo en cuenta tanto las limitaciones actuales como los futuros desarrollos de las tecnologías web, en particular la emergente Web semántica[1].

En general, el filtrado de información concierne el procesamiento de una cantidad determinada de datos, a fin de devolver solo aquellos parámetros satisfactorios. Aunque esta noción precede al nacimiento de Internet, el éxito y la difusión de los servicios, tales como el correo electrónico y la Web, se tradujo en la necesidad de regular y controlar el tráfico de red y de impedir el acceso, transmisión y entrega de información no deseada.

En la actualidad, el filtrado de información se aplica a varios niveles y servicios de la arquitectura TCP/IP (acrónimo de los términos ingleses *Transmission Control Protocol/Internet Protocol*) [2]. Dos ejemplos típicos son el filtrado de *spam* y el filtrado de red. Las estrategias adoptadas son diversas, así como la obtención, en la mayoría de los casos, de un servicio eficaz y eficiente. Sin embargo, el filtrado de datos en línea (texto, imágenes, vídeo y audio) sigue siendo un problema complejo cuando se requiere la evaluación de su significado semántico, para verificar si cumplen determinados requisitos. La razón es que las técnicas disponibles no permiten una representación exacta y precisa de los contenidos multimedia. Para los servicios como los motores de búsqueda, esto se traduce en una gran cantidad de información inservible devuelta como resultado de una consulta. El problema es mucho más grave cuando hemos tenido que prevenir a los usuarios sobre la recuperación de recursos con un contenido determina-

[1] La Web semántica (Berners-Lee *et al*., 2001) propone superar las limitaciones de la Web actual mediante la introducción de descripciones explícitas del significado, la estructura interna y la estructura global de los contenidos y servicios disponibles en la WWW.

[2] La arquitectura TCP/IP permite a cualquier clase de ordenador, aún empleando sistemas operativos diferentes, comunicarse entre sí.(Stevens y Wright, 1994). Consta de 4 capas: (i) capa de enlace, (ii) capa de red, (iii) capa de transporte y (iv) capa de aplicación.

do (por ejemplo, porque un usuario no tiene los derechos para acceder a él o porque el contenido es inapropiado para el usuario que lo solicita). En tal caso, el filtrado debe basarse en una descripción exhaustiva de recursos con el fin de evaluarlos correctamente.

El filtrado web es una materia bastante complicada, ya que los requisitos del mismo no han sido abordados a fondo hasta el momento. Particularmente, los sistemas de filtrado web disponibles se centran en dos objetivos principales: (i) la protección y (ii) el rendimiento. Por un lado, el filtrado debe evitar a toda costa la información dañina a la que se tiene acceso. Es decir, es preferible bloquear el contenido adecuado en lugar de permitir visualizar contenidos inadecuados. Por otro lado, el procedimiento de filtrado no debe afectar al tiempo de retardo en respuesta; es decir, la evaluación de una solicitud de acceso y el resultado obtenido deben ser realizados con rapidez.

#### 2.2.1.1 **Enfoques de filtrado web y valoración**

El filtrado web conlleva evitar que los usuarios tengan acceso a los recursos web con contenido inapropiado. Como tal, se ha convertido en una necesidad apremiante para los gobiernos y las instituciones, debido al enorme crecimiento de la Web durante los últimos años y la gran variedad de usuarios que tienen acceso y hacen uso de ella. En este contexto, a pesar de que su principal objetivo es la protección de los menores de los contenidos dañinos en línea (como por ejemplo, la pedofilia, la pornografía y la violencia), el filtrado web ha sido y es considerado por las instituciones, como empresas, bibliotecas y escuelas, un medio para evitar el uso indebido de sus servicios de red y recursos.

Esta clase de filtrado implica dos cuestiones principales: (i) la valoración y (ii) el filtrado. La valoración se refiere a la clasificación (y posiblemente al etiquetado) de contenido en los sitios web, desde el punto de vista del filtrado, y se puede realizar manual o automáticamente, ya sea por organizaciones de terceros (valoraciones de terceros) o por los mismos propietarios del sitio web (autovaloración). La valoración del sitio web no se suele realizar sobre la marcha, como una petición de acceso, ya que aumentaría dramáticamente el tiempo de respuesta. Por otra parte, la cuestión del filtrado es aplicada por los sistemas de filtrado (mecanismos capaces de gestionar las solicitudes de acceso a los sitios Web) para permitir o denegar el acceso a documentos en línea sobre la base de un conjunto dado de políticas. Estas políticas denotan qué usuarios pueden o no acceder al contenido en línea y a las calificaciones relacionadas con el recurso web solicitado. Los sistemas de filtrado pueden ser, o bien basados en el

cliente o bien en el servidor y hacer uso de una variedad de técnicas de filtrado.

Actualmente, los servicios de filtrado son proporcionados por los proveedores de servicios de Internet (ISP, de las voces inglesas *Internet Service Provider*), empresas TIC (acrónimo de Tecnologías de la Información y la Comunicación) y organizaciones sin ánimo de lucro, los cuales valoran sitios web y realizan varios sistemas de filtrado disponibles para los usuarios. De acuerdo con la estrategia más comúnmente utilizada, los sitios web se han valorado de forma manual o automática (mediante técnicas basadas en redes neuronales) y se dividen en un conjunto predefinido de categorías. Los abonados al servicio pueden seleccionar a qué categorías de sitios web no desean tener acceso. Con el fin de simplificar esta tarea, los servicios ofrecidos de filtrado proporcionan acceso personalizado a la Web, según los cuales algunas categorías de los sitios web se consideran inadecuados, por defecto, para ciertas categorías de usuarios. Este principio también es adoptado por algunos motores de búsqueda (como *Google SafeSearch*[1]) que devuelven sólo los sitios web pertenecientes a las categorías consideradas apropiadas.

A continuación, hemos descrito un método de filtrado (Bertino *et al.*, 2006) que tiene por objetivo mejorar, ampliar y hacer más flexibles las técnicas disponibles mediante la aplicación de dos principios fundamentales:

1. El método de filtrado deberá proporcionar soporte a diferentes técnicas de calificación y filtrado (tanto directas como indirectas), por lo que pueden ser utilizadas individualmente o en combinación, de acuerdo con las necesidades de los usuarios.

2. Las características de los usuarios deben ser descritas con precisión, a fin de proporcionar un filtrado más eficaz y flexible.

A partir de estos principios, los autores definen un modelo formal para el filtrado web: MSFM (acrónimo de los términos ingleses *Multi-Strategy Filtering Model*). El objetivo que persiguen es diseñar un marco general de filtrado, para abordar tanto la flexibilidad y las cuestiones de protección, lo que posiblemente puede ser personalizado según las necesidades de los usuarios mediante el uso de sólo un subconjunto de sus características.

Los usuarios y los recursos a los que se aplica una política se indican de forma explícita mediante la especificación de sus identificadores, o implícitamente mediante la especificación de las restricciones en sus propiedades (por ejemplo,

[1] `http://www.google.com/preferences`

todos los usuarios cuya edad es menor de 16 años no pueden acceder a los recursos de contenido pornográfico). Emplean dos tipos de especificaciones: (i) explícitas e (ii) implícitas. Estas especificaciones son abstracciones de las estrategias de filtrado adoptadas: las especificaciones explícitas corresponden a los enfoques basados en listas (es decir, listas blancas o listas negras), mientras que las especificaciones implícitas combinan todas las estrategias que se basan en las calificaciones, como por ejemplo: PICS (acrónimo del inglés *Platform for Internet Content Selection* (Resnick y Miller, 1996)) y / o categorías de contenido (por ejemplo, *RuleSpace*, una herramienta adoptada por *Yahoo*[1] y otros ISPs para sus servicios de control parental[2]). Esta integración de especificaciones del sistema extrínseco difiere totalmente de enfoques como el basado en el modelo ABC (Lagoze y Hunter, 2001), con el objetivo de proporcionar la interoperabilidad semántica entre ontologías.

### 2.2.2 El filtrado de textos

El diseño de sistemas de filtrado de textos beneficia a la investigación en recuperación de información, modelado de usuarios, sistemas de recomendación, aprendizaje automático y otros campos. Sin embargo, el filtrado de textos es un proceso único en la búsqueda de información que se distingue por ser un enfoque que satisface, relativamente, los intereses del usuario en documentos que contienen texto.

Los sistemas de filtrado de textos deben desarrollar representaciones de los documentos y de los intereses de los usuarios, también deben estar previstos de algún medio para comparar los documentos con los intereses y deben poseer algún procedimiento que utilice los resultados de esas comparaciones para ayudar al usuario con la detección de documentos. La investigación en recuperación de textos ha producido una serie de contenidos (basados en las representaciones) que utilizan la frecuencia con la que los términos aparecen en los documentos, mientras que la evolución en el campo de los sistemas de recomendación está produciendo un conjunto complementario de características basado en anotaciones compartidas de otros usuarios.

Por su parte, el autor Oard (1997) argumenta que una combinación sinérgica de los 2 enfoques anteriores (recuperación de textos y sistemas de recomendación), ofrece la posibilidad de un mayor rendimiento que uno u otro enfoque puede llegar a producir de manera aislada. Cuando se combina con una res-

[1] `http://www.yahoo.com/`

[2] `http://www.symantec.com/popup.jsp?popupid=parental_controls`

puesta implícita o explícita por parte del usuario sobre los documentos que se han examinado, las representaciones de textos proporcionan una base para la construcción de los perfiles que representarán los intereses del usuario. El trabajo existente en la retroalimentación implícita muestra esperanza, y mediante el aprovechamiento de esa promesa puede ser posible mejorar significativamente el rendimiento del componente de cooperación de un sistema integrado de filtrado de textos. Con una base fértil para el futuro desarrollo, el autor asegura que los futuros sistemas de filtrado de textos explorarán aspectos interesantes y encontrarán aplicaciones en áreas que son críticas, para el uso eficaz de la infraestructura de información global emergente.

El objetivo de la clasificación del textos es proporcionar una estructura a un repositorio desestructurado de textos, facilitando así el almacenamiento, búsqueda y navegación, según Sebastiani (2006). Para ello, se definen una serie de tareas citadas por Gómez Hidalgo *et al*. (2009), las cuales se han incrementado durante los años. Estas tareas se pueden clasificar utilizando dos ejes: tipo de aprendizaje (desarrollado más profundamente en la sección 4.1) y la granularidad de los elementos del texto.

- Los 2 tipos de aprendizaje son definidos por el conjunto de entrenamiento de control:

    1. **Aprendizaje supervisado.** Conjunto de clases que se conocen a la hora de construir el conjunto de entrenamiento y disponen de ejemplos para cada una de las clases.
    2. **Aprendizaje no supervisado (*clustering*).** Conjunto de clases que no se conocen antes del entrenamiento y cuyo objetivo es agrupar entidades textuales acorde a un contenido similar.

- Se pueden definir 3 niveles de granularidad, considerando términos, frases o documentos como elementos atómicos:

    1. **Términos.** Engloban desde la raíz de la palabra y palabras sueltas hasta las expresiones cortas.
    2. **Frases.** Contienen desde las oraciones hasta las frases complejas.
    3. **Documentos.** Incluyendo pequeños correos no deseados, artículos de tamaño medio o incluso libros enteros.

Algunas veces, estas tareas no son el objetivo principal de un sistema de categorización de texto, y pueden actuar como un servicio para otras tareas. Por

ejemplo, el etiquetado de POS (acrónimo de las voces inglesas *Part of Speech*), el reconocimiento de entidades identificadas y la desambiguación, son utilizadas a menudo como pasos previos a la categorización de textos para mejorar la calidad de los atributos, asignando un significado adecuado a un término.

Actualmente, existen diferentes modelos para poder categorizar y filtrar textos en Internet. Como por ejemplo en Hidalgo *et al*. (2003), en donde la representación más utilizada es el modelo de espacio vectorial (VSM, de los términos ingleses *Vector Space Model*), a la vez que aplica una lista de palabras vacía, cuando el lenguaje puede soportarlo (normalmente lenguajes occidentales). Por otra parte, también aplican diferentes técnicas como la ganancia de información (IG, de las voces inglesas *Information Gain*) que se utiliza para filtrar aquellos términos que no parecen ser importantes para la detección, así como diferentes algoritmos de aprendizaje automático como: el clasificador bayesiano ingenuo (del inglés *Naïve Bayes*), los árboles de decisión (de los términos ingleses *Decision Trees*) o las máquinas de soporte vectorial (SVM, acrónimo del inglés *Support Vector Machines*). Todo este proceso se desarrolla más profundamente en los siguiente capítulos.

## 2.3 Métodos de reputación y relevancia

El concepto de reputación está estrechamente relacionado con el concepto de fiabilidad, pero es evidente que hay una diferencia clara e importante. A efectos de este estudio, hemos definido la reputación de acuerdo con el Concise Oxford Dictionary[1]:

> «*La reputación es lo que generalmente se dice o se cree acerca del carácter o prestigio de una persona o cosa.*»

Esta definición se corresponde bien con la opinión de los investigadores de redes sociales Freeman (1979); Marsden y Lin (1982), que afirman que la reputación es una magnitud derivada de la red social subyacente, la cual es totalmente visible para todos los miembros de la red. La diferencia entre la confianza y la reputación se ilustra con los siguientes enunciados:

1. “Yo confío en tí debido a tu buena reputación.”
2. “Yo confío en tí a pesar de tu mala reputación.”

[1] `http://oxforddictionaries.com`

Suponiendo que las 2 frases se refieren a transacciones idénticas, la primera declaración refleja que la parte que confía es consciente de la reputación de la otra parte y basa su confianza en ello. La segunda declaración, refleja que la parte que confía tiene algún conocimiento privado sobre la otra parte. Por ejemplo, mediante factores como la experiencia directa o relación íntima, se puede anular cualquier reputación que una persona pueda tener. Esta observación muestra que la confianza, en última instancia, es un fenómeno personal y subjetivo que se basa en varios factores o evidencias y que algunos de éstos tienen más peso que los demás. Por lo general, la experiencia personal tiene más peso que las referencias de otras personas, en cuanto a confianza o reputación se refiere, pero en ausencia de la experiencia personal, la confianza a menudo tiene que estar basada en las referencias de los demás.

La reputación puede ser considerada como una medida colectiva de confianza (en el sentido de la fiabilidad), basada en las referencias o calificaciones de los miembros de una comunidad. La confianza subjetiva de un individuo puede ser derivada de una combinación de referencias recibidas y la experiencia personal. Con el fin de evitar la dependencia y los bucles, se requiere que las referencias se basen, solamente, en la experiencia de *primera mano*, y no en otras referencias de segundas o terceras personas. Como consecuencia, una persona solo debe dar una referencia subjetiva de confianza cuando está basada en un hecho de *primera mano* o cuando el hecho procedente de una *segunda mano* ha sido suprimido (Jøsang y Pope, 2005). Es posible abandonar este principio, por ejemplo, cuando el peso de la referencia de confianza es normalizado o dividido por el número total de referencias dado por una sola entidad. Y este último principio se aplica en el algoritmo de *PageRank* de *Google*[1] (Page *et al.*, 1999).

La reputación puede relacionar a una persona o a un grupo. La reputación de un grupo puede, por ejemplo, ser modelada como el promedio de todas las reputaciones de sus miembros individualmente o como el promedio de cómo el grupo es percibido como un todo por las partes externas. El estudio de Tadelis (2003) muestra que un individuo perteneciente a un determinado grupo heredará una gran reputación basada, a *priori*, en la reputación de ese grupo. Si el grupo es respetable, todos sus miembros individuales, a *priori*, serán percibidos como respetables y viceversa.

[1] `http://www.google.com`

### 2.3.1 Los motores computacionales de reputación

Normalmente, los sistemas de reputación se basan en información pública (información de *segunda mano* a disposición de todo el mundo), con el fin de reflejar la opinión de la comunidad. La parte que confía en la reputación de otra parte remota, está confiando en dicha parte mediante la confianza transitiva (Jøsang y Pope, 2005).

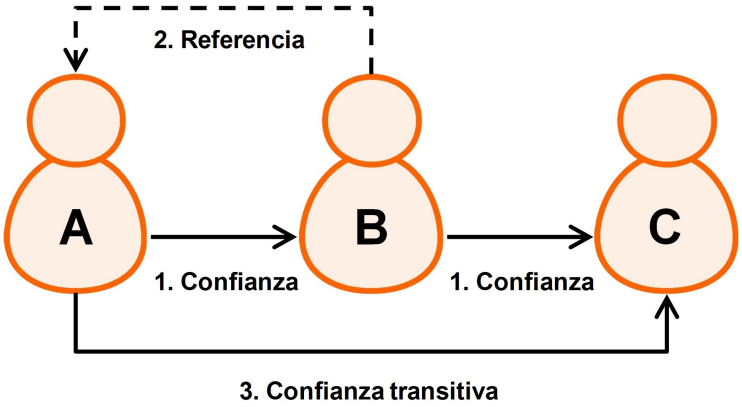



**Figura 2.1:** Ejemplo del principio de confianza transitiva.

Como se muestra en la Figura 2.1, la confianza transitiva es un sistema de 3 elementos participantes, que a su vez también se realiza en 3 pasos. En el primer paso, se muestra como el sujeto A confía en el sujeto B y a su vez el sujeto B confía en el sujeto C, sin conocerse mutuamente el sujeto A y el C entre ellos. En el segundo paso, debido a la confianza existente entre los sujetos A y B, el sujeto B referencia al sujeto A sobre el sujeto C, ya que confía en la reputación de dicho sujeto. Por último, ya que el sujeto A confía en la reputación del sujeto B, el sujeto A comenzará a confiar en el sujeto C, con lo que hemos obtenido lo denominado confianza transitiva.

En cambio, algunos sistemas toman como entrada tanto la información pública como la privada. En cuanto a la información privada, hemos destacado que es más fiable que la información pública. Un ejemplo sería el resultado de la experiencia personal de cada usuario.

#### 2.3.1.1 La suma o promedio de las calificaciones

La forma más simple de procesar las puntuaciones de reputación es sumando el número de calificaciones positivas y calificaciones negativas por separado, y

mantener una puntuación total como la puntuación positiva menos la puntuación negativa. Este es el principio utilizado en el foro de reputación de *eBay*[1] (Resnick y Zeckhauser, 2002). La ventaja es que cualquiera puede entender el principio existente detrás de la computación. La desventaja es que es primitiva y por lo tanto, ofrece una pobre imagen sobre el resultado de la reputación de los participantes, aunque esto también es debido a la manera en que se proporciona la puntuación. Un esquema un poco más avanzado proponen Schneider *et al*. (2000), que calculan la puntuación de reputación como el promedio de todas las calificaciones. Este principio es usado en los sistemas de reputación de numerosos sitios webs comerciales como *Epinions*[2] o *Amazon*[3]. Los modelos avanzados en esta categoría calculan un promedio ponderado de todas las calificaciones, donde el peso de la calificación puede ser determinado por factores como la fiabilidad o reputación, la antigüedad del voto, la distancia entre la calificación y la puntuación actual, etc.

#### 2.3.1.2 **Los sistemas bayesianos**

Los sistemas bayesianos toman como entrada calificaciones binarias (por ejemplo, positivo o negativo) y procesan las puntuaciones de reputación mediante la actualización de una variable estadística de distribución de probabilidad beta. El resultado de la reputación, a *posteriori*, se calcula combinando el resultado de la reputación, a *priori*, con la nueva calificación (Jsang y Ismail, 2002; Mui *et al*., 2001a,b, 2002; Whitby *et al*., 2005).

Las puntuaciones de reputación pueden ser representadas como una función de densidad de probabilidad beta (PDF, acrónimo de las voces inglesas *Probability Density Functions*) de parámetros $\alpha$ y $\beta$, donde $\alpha$ y $\beta$ representan la cantidad de calificaciones positivas y negativas respectivamente. La ventaja de los sistemas bayesianos es que proporcionan una base teórica sólida para las puntuaciones de reputación procesadas y la única desventaja es que podrían ser demasiado complejos entenderlos.

La familia de distribuciones beta es una familia de funciones de distribución indexada por los 2 parámetros $\alpha$ y $\beta$. La función beta PDF denotada por beta($p|\alpha, \beta$) puede expresarse usando la función *Gamma* $\Gamma$[4] como:

[1] `http://ebay.com`
[2] `http://www.epinions.com`
[3] `http://www.amazon.com`
[4] La función *Gamma* generaliza el factorial para cualquier valor complejo de $n$.

$$beta(p|\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1-p)^{\beta-1} \text{ donde } 0 \leq p \leq 1; \alpha,\beta > 0 \qquad (2.1)$$

con la restricción de que la variable probabilística $p \neq 0$ si $\alpha$ <1 y $p \neq 1$ si $\beta$ <1. La ecuación 2.1 hemos podido representarla con factoriales, dado que en nuestro caso los parámetros $\alpha$ y $\beta$ son números enteros:

$$beta(p|\alpha,\beta) = \frac{(\alpha+\beta)!}{\alpha!\beta!} p^{\alpha-1}(1-p)^{\beta-1} \text{ donde } 0 \leq p \leq 1; \alpha,\beta > 0 \qquad (2.2)$$

Una vez expresada la función beta PDF, se emplea la esperanza matemática[1] para poder obtener el valor esperado de la distribución beta:

$$E(p) = \int_0^1 p \cdot beta(p|\alpha,\beta)dp = \frac{\alpha}{(\alpha+\beta)} \qquad (2.3)$$

Cuando no hemos conocido ningún valor, la distribución, a *priori*, es una distribución beta PDF uniforme con $\alpha = 1$ y $\beta = 1$. Luego, después de obtener los resultados $r$ positivo y $s$ negativo, la distribución, a *posteriori*, es la beta PDF con $\alpha = r + 1$ y $\beta = s + 1$.

Una función PDF de este tipo expresa la probabilidad incierta de que las interacciones del futuro serán positivas. Lo más natural es definir la puntuación de reputación en función de la esperanza matemática. Por ejemplo, la función beta PDF con los siguientes valores: (i) 7 como resultado positivo y (ii) 1 como resultado negativo, su esperanza matemática probabilística acorde a la ecuación 2.3 es $E(p) = 0{,}8$. Este resultado puede ser interpretado como que la frecuencia relativa de un resultado positivo es algo incierto en el futuro y que su valor más probable es 0,8.

#### 2.3.1.3 **Los modelos discretos de confianza**

A menudo, los seres humanos somos capaces de valorar mejor el rendimiento en forma de declaraciones verbales discretas, que mediante medidas continuas. Esto también es válido para la determinación de medidas de confianza; y algunos autores, entre ellos Abdul-Rahman y Hailes (2000); Cahill *et al*. (2003);

[1] La esperanza matemática representa el concepto de media ponderada para distribuciones continuas.

Carbone *et al*. (2003); Manchala (1998), han propuesto modelos discretos de confianza.

En el modelo que presentan Abdul-Rahman y Hailes (2000), la fiabilidad de un agente $X$ puede ser contemplada como *Muy Fiable*, *Fiable*, *Poco Fiable* o *Muy Poco Fiable*. La parte confiadora puede aplicar su percepción sobre la fiabilidad de la otra parte, antes de tener en cuenta la referencia del agente $X$. Esta referencia marcará la confianza derivada en dicho agente y podrá ser determinada en las tablas de consulta. Cada vez que la parte confiadora ha tenido cierta experiencia personal con el agente $X$, ésta puede ser usada para determinar la fiabilidad de la tercera parte a confiar. La hipótesis de los autores es que la experiencia personal refleja la fiabilidad real respecto al agente $X$ y que las referencias sobre el agente $X$ que difieren de la experiencia personal, indicarán si la tercera parte a confiar es subestimada o sobrestimada.

La desventaja de las medidas discretas es que no se prestan fácilmente a los principios computacionales; por lo cual, se deben de utilizar mecanismos heurísticos, como las tablas de consulta.

#### 2.3.1.4 **Los modelos de flujo**

Los sistemas que calculan la confianza o reputación por iteración transitiva a través de largas cadenas iterativas o arbitrarias, pueden denominarse modelos de flujo. Algunos de estos modelos asumen un peso constante de confianza o reputación para toda la comunidad, y éste puede ser distribuido entre todos los miembros de la comunidad. Los participantes solo pueden incrementar su reputación o confianza a costa de los demás. El *PageRank* de *Google*, descrito en la sección 2.3.2.4, el algoritmo *Appleseed* (Ziegler y Lausen, 2004) y el esquema de reputación de *Advogato*[1] (Levien, 2002), pertenecen a esta categoría.

En general, la reputación de un participante se incrementa como una función de flujo entrante, mientras que disminuye como una función de flujo saliente. En el caso de *Google*, muchos enlaces a una página web contribuyen a incrementar el *PageRank*, mientras que muchos enlaces desde una página web contribuyen a la disminución del *PageRank* de dicha página.

Los modelos de flujo no siempre requieren que la suma de las puntuaciones de confianza o reputación sea constante. Un ejemplo es el modelo *EigenTrust* (Kamvar *et al*., 2003), el cual calcula las puntuaciones de confianza de un agen-

[1] `http://www.advogato.org/trust-metric.html`

te en redes P2P[1] (acrónimo de las voces inglesas *Peer-to-Peer*) mediante multiplicaciones repetidas e iterativas y agrega las puntuaciones de confianza a lo largo de cadenas transitivas hasta que dichas puntuaciones convergen a valores estables, para todos los agentes miembros de la comunidad P2P.

### 2.3.2 Los sistemas de reputación comerciales

#### 2.3.2.1 El foro *Feedback* de *eBay*

*eBay* es un popular sitio de subastas que permite a los vendedores listar los artículos para venderlos y a los compradores pujar por dichos elementos. El llamado foro *Feedback* en *eBay* ofrece al comprador y al vendedor la oportunidad de votarse entre sí como positivo, negativo o neutro (por ejemplo, 1, -1, 0) después de completar la transacción. Los compradores y vendedores también tienen la posibilidad de dejar comentarios tanto positivos como negativos.

El foro *Feedback* es un sistema centralizado de reputación, en donde *eBay* recoge todas las calificaciones y calcula los resultados. El resultado total de la reputación de cada participante es la suma de las calificaciones positivas (de usuarios únicos) menos la suma de las calificaciones negativas (también de usuarios únicos). Con el fin de proporcionar información sobre el comportamiento más reciente de un participante, se muestra el total de votos positivos, negativos y neutros en tres ventanas temporales diferentes: (i) últimos seis meses, (ii) último mes y (iii) últimos siete días.

Existen muchos estudios empíricos sobre el sistema de reputación de *eBay* (Resnick *et al.*, 2006). En general, las calificaciones observadas en *eBay* son sorprendentemente positivas, debido a que los compradores dan calificaciones a los vendedores el 51,7 % de las veces y los vendedores ofrecen calificaciones a los compradores el 60,6 % (Resnick y Zeckhauser, 2002). De todas las calificaciones ofrecidas, menos del 1 % es negativa, menos del 0,5 % es neutra y aproximadamente el 99 % es positiva. También se encuentra una gran correlación entre las calificaciones de los compradores y vendedores, lo que sugiere que hay un grado de reciprocidad en las calificaciones positivas y represalia en las calificaciones negativas. Esto sería problemático si la obtención de calificaciones honestas y justas se convirtiera en una meta. Por lo que una posible solución podría ser no dejar que los vendedores califiquen a los compradores.

El sistema de reputación de *eBay* es muy primitivo y puede ser engañoso.

[1] Una red P2P es una red de ordenadores, en la que todos los nodos se comportan como iguales entre sí, sin la aparición de servidores o clientes.

Con unas pocas calificaciones negativas, un participante con 100 positivas y 10 negativas debería aparecer mucho menos respetable que un participante con 90 positivas y ninguna negativa, pero en *eBay* tendrían la misma puntuación total de reputación. A pesar de sus inconvenientes y su naturaleza primitiva, el sistema de reputación de *eBay* parece tener un fuerte impacto positivo en el propio *eBay* como mercado. Cualquier sistema que facilita la interacción entre los seres humanos, depende de cómo ellos responden al mismo, y parece que la gente responde satisfactoriamente al sistema de *eBay* y a su componente de reputación.

#### 2.3.2.2 *Amazon*

*Amazon* es sobre todo una tienda en línea que permite a sus miembros (cualquiera puede convertirse en miembro simplemente registrándose) escribir comentarios sobre sus productos. Los comentarios constan de texto y de una calificación entre 1 y 5. El promedio de todas las calificaciones da a un producto su calificación media. Los usuarios, incluidos los no miembros, pueden votar en los comentarios como *útil* o *no útil*; y el número total de *útiles*, así como el número total de votos, se muestran en cada comentario. El orden en que los comentarios se enumeran, puede ser elegido por el usuario de acuerdo a criterios como: (i) *el más reciente primero*, (ii) *el más útil primero* o (iii) *la calificación más alta primero*.

Otro elemento importante en este sistema es la figura del crítico. *Amazon* determina el *ranking* de cada crítico en función del número de votos útiles que cada crítico ha recibido, así como otros parámetros no revelados públicamente. Los críticos que se encuentran entre los 1.000 críticos con puntuación más alta, se les asigna el estado de *Top 1.000*, *Top 500*, *Top 100*, *Top 50*, *Top 10* o crítico *#1*. *Amazon* tiene un sistema de *gente favorita*, donde cada miembro puede elegir a otros miembros como críticos favoritos y listar cuántos miembros tienen a un crítico específico listado como persona favorita. Ambas acciones también influyen en el rango de dicho crítico. Aparte de otorgar a algunos miembros el estado de críticos principales, *Amazon* no ofrece ningún incentivo financiero; sin embargo, existen obviamente otros incentivos financieros externos a *Amazon* que pueden desempeñar un papel importante. Por ejemplo, es fácil imaginar por qué los editores quieren pagar a la gente para escribir buenos comentarios acerca de sus libros en *Amazon*.

Hay muchos informes sobre ataques contra el sistema de comentarios de *Amazon*, donde los distintos tipos de *pucherazo* han elevado a ciertos críticos

a crítico superior, o diferentes tipos de *formas mal habladas* han destronado a críticos superiores. Esto no es sorprendente, debido al hecho de que los usuarios pueden votar sin convertirse en miembro. Por ejemplo, el crítico número 1 de *Amazon*, suele ser alguien que publica más comentarios que cualquier persona viva, en el caso de que se requiriese que la persona leyera cada libro. Lo que indica que es el esfuerzo de un grupo de personas, presentado como el trabajo de una sola, el necesario para llegar al *top*. *Amazon* solo permite un voto por *cookie*[1] registrada para cualquier comentario dado; sin embargo, el borrar esa *cookie* o cambiar a otro ordenador, permitirá al mismo usuario votar en el mismo comentario una vez más. Siempre habrá nuevos tipos de ataques y *Amazon* necesita estar en alerta y responder a los nuevos ataques que surjan. Por otra parte, debido a la vulnerabilidad del esquema de comentarios, no puede ser descrito como un esquema robusto.

#### 2.3.2.3 ***Slashdot***

*Slashdot*[2] comenzó en 1997 como un foro para la publicación de artículos y de comentarios sobre los propios artículos. Cuando la comunidad de usuarios era reducida, la gestión de los artículos y comentarios suponía una tarea hasta cierto punto fácil; pero pronto, el número de usuarios registrados comenzó a crecer de forma exponencial. De forma paralela, también empezó a crecer el *spam* entre los artículos y comentarios, por lo que se convirtió en un objetivo necesario y prioritario la introducción de moderación en los contenidos. En un principio, se creó un grupo formado por 25 personas que se encargaban de gestionar los artículos y comentarios con potestad para suprimir los contenidos inadecuados. Pero este número tuvo que incrementarse hasta los 400, con el fin de intentar dar respuesta al continuo crecimiento de nuevos miembros.

Posteriormente, se decidió crear un esquema de moderación automático formado por dos capas básicas: (i) la capa M1, encargada de moderar los comentarios de los artículos y (ii) la capa M2, la cual se ocupa de moderar la capa M1. Los artículos publicados en *Slashdot* son seleccionados según el criterio del personal del propio sitio web, basándose en las propuestas de la comunidad *Slashdot*. Una vez publicado un artículo, cualquiera puede hacer comentarios sobre ese artículo.

Los usuarios de *Slashdot* pueden iniciar sesión simplemente registrándose o permanecer anónimos navegando por la Web. Debido a ello, la lectura de libros y

[1] Una *cookie* es un pequeño archivo con información almacenada por algunos sitios web.
[2] `http://slashdot.org`

comentarios, así como la escritura de comentarios sobre artículos puede ser anónima. Como cualquier usuario puede escribir comentarios, éstos necesitan ser moderados y únicamente los usuarios registrados son los idóneos para convertirse en moderadores. Con cierta regularidad, habitualmente cada 30 minutos, *Slashdot* automáticamente selecciona un grupo de moderadores de tipo M1 entre los usuarios registrados con mayor antigüedad y ofrecen a cada moderador 3 días para gastar cierto número de puntos de moderación (normalmente 5 puntos). Cada punto de moderación puede ser gastado moderando un comentario, dándole una calificación seleccionada de una lista de adjetivos negativos (*offtopic*, *flamebait*, *troll*, *redundant*, *overrated*) o positivos (*insightful*, *interesting*, *informative*, *funny*, *underrated*). Por cada comentario se mantiene una puntuación entera entre los valores -1 y 5. La puntuación inicial es 1 pero puede variar según diferentes situaciones: (i) la valoración positiva de un moderador causa el incremento de 1 punto en la puntuación del comentario y (ii) la valoración negativa por parte del moderador causa el decremento de 1 punto.

El propósito de la puntuación en los comentarios es ser capaz de filtrar los comentarios buenos de los malos y permitir a los usuarios establecer umbrales cuando estén leyendo artículos o publicando comentarios en *Slashdot*. Un usuario que solamente quiere leer los mejores comentarios, puede establecer el umbral al valor 5; mientras que un usuario que quiere leerlos todos, puede establecerlo al valor -1.

El sistema de reputación de *Slashdot* reconoce que el *gusto* de un moderador puede influir en cómo él o ella califican un comentario. El tener un conjunto de calificaciones positivas y uno de calificaciones negativas está dirigido a resolver este problema, cada conjunto con sus diferentes tipos de *gusto* dependiendo de las opciones de valoración. La idea es que los moderadores con diferentes *gustos* puedan dar diversas valoraciones a un comentario meritorio, pero cada calificación seguirá siendo positiva. Similarmente, los moderadores con *gusto* diferente pueden dar distintas valoraciones a un comentario no meritorio, pero todas las valoraciones seguirán siendo negativas. El personal de *Slashdot* también es capaz de gastar cantidades arbitrarias de los puntos de moderación, los cuales hacen que estas personas sean omnipotentes y por lo tanto, capaces de estabilizar manualmente el sistema en caso de que fuese atacado por volúmenes elevados de *spam* y calificaciones injustas.

El sistema de reputación de *Slashdot* dirige y estimula el enorme esfuerzo de colaboración al moderar miles de publicaciones cada día. El sistema está constantemente siendo afinado y modificado, y se puede describir como un experimento en curso en la búsqueda de la mejor manera práctica de promover

publicaciones de calidad, disuadir el ruido y hacer de *Slashdot* un sitio legible y útil para una gran comunidad, en la medida de lo posible.

#### 2.3.2.4 El sistema de *ranking* de páginas web de *Google*

El *PageRank* propuesto por Page *et al*. (1999) representa una manera de clasificar los mejores resultados de búsqueda basados en la reputación de una página. Hablando en términos generales, *PageRank* clasifica una página acorde al número de páginas que apuntan hacia ella. Esto puede ser descrito como un sistema de reputación, ya que la colección de enlaces de una página dada, puede ser vista como información pública que, a su vez, puede ser combinada para obtener una puntuación de reputación. Un único enlace a una página web determinada puede ser considerado como un voto positivo de esa página web. El motor de búsqueda de *Google* está basado en el algoritmo de *PageRank*. El rápido aumento de la popularidad de este buscador web fue causado por los mejores resultados de búsqueda que el algoritmo entregaba. Su definición es la siguiente:

> «*Sea $P$ un conjunto de páginas web enlazadas y sean "u" y "v" páginas web en $P$. Se denota $N^-(u)$ como el conjunto de páginas web que apuntan a "u" y dado $N^+(v)$ como el conjunto de páginas web a las que "v" apunta. Sea $E$ un vector sobre $P$ correspondiente a una fuente del rango. Entonces, el PageRank de una página web "u" es:*
>
> $$R(u) = cE(u) + c \sum_{v \in N^-(u)} \frac{R(v)}{|N^+(v)|} \tag{2.4}$$
>
> *donde "c" es elegido de tal manera que $\sum_{u \in P} R(u) = 1$*»

Los autores Page *et al*. (1999) recomiendan que $E$ sea elegido de tal manera que $\sum_{u \in P} E(u) = 0{,}15$. El primer término $cE(u)$ en la ecuación 2.4 nos ofrece el valor de clasificación basado en la clasificación inicial. El segundo término $c \sum_{v \in N^-(u)} \frac{R(v)}{|N^+(v)|}$, nos ofrece el valor de la clasificación en función de los enlaces que apuntan a "$u$".

De acuerdo con la ecuación 2.4, $R \in [0, 1]$; pero los valores de *PageRank* que se proporciona al público son escalados entre 0 y 10 con incrementos de 0,25. Se puede indicar el *PageRank* de una página $u$ como $PR(u)$. Esta medida pública puede ser vista por cualquier página web utilizando la barra de herramientas de *Google*. Aunque no se especifica exactamente cómo se calcula el algoritmo,

el vector fuente $E$ puede ser definido en todas las páginas web de todos los dominios ponderados mediante el coste de compra de cada nombre del dominio. Asumiendo que la única manera de mejorar el *PageRank* de una página es comprando nombres de dominio, Clausen (2004) muestra que existe un límite mínimo para el coste de obtener un valor de *PageRank* bueno.

Sin especificar muchos detalles, *Google* afirma que el algoritmo de *PageRank* que están usando también tiene en cuenta otros elementos, con el propósito de hacer que sea difícil o caro el influir deliberadamente en él.

Con el fin de proporcionar una interpretación semántica de un valor de *PageRank*, un enlace puede ser visto como una referencia positiva a la página que apunta. Las referencias negativas no existen, por lo que es imposible hacer una lista negra de páginas web con el algoritmo 2.4. Antes de que *Google* entrara en el mercado de los motores de búsqueda, algunos administradores web promovieron sitios en forma de *spam*, llenando las páginas web con grandes cantidades de palabras clave de búsqueda comúnmente usadas como texto invisible o metadatos. Su finalidad era que la página tuviese una alta probabilidad de ser recogida por un motor de búsqueda, sin importar lo que el usuario hubiera buscado. Aunque este suceso todavía puede ocurrir, el algoritmo parece que ha reducido este problema: se necesita un alto *PageRank*, además de coincidir las palabras clave, para que una página sea presentada al usuario.

En el *PageRank* se aplica el principio de confianza transitiva, explicado en la Figura 2.1, debido a que los valores del rango pueden fluctuar a través de largas cadenas de hipervínculos enlazados. Algunos modelos teóricos, como los presentados en (Kamvar *et al.*, 2003; Levien, 2002; Ziegler y Lausen, 2004), también permiten la transitividad enlazada y/o infinita.

## 2.4 Series temporales y predicción

El análisis de series temporales es el proceso de emplear técnicas estadísticas para modelar y explicar series de datos dependientes del tiempo. Por otra parte, la predicción de series temporales es el proceso de utilización de un modelo para generar predicciones para hechos futuros basados en hechos conocidos del pasado.

Las series temporales de datos tienen un orden temporal natural, esto difiere de las típicas aplicaciones de minería de datos o aprendizaje automático en las que cada dato es un ejemplo independiente del concepto que hay que aprender, y el orden de los datos dentro de un conjunto de datos, no impor-

ta. Ejemplos de aplicaciones de series temporales son: estadística de señales de pre-procesamiento y comunicaciones, procesamiento de control industrial, la econometría, la meteorología, la física, la biología, la medicina, la oceanografía, la sismología, la astronomía, la política y la psicología, como se analizan en Pollock *et al*. (1999).

En estas áreas es necesario una gran velocidad computacional, incluyendo el modelado y la simulación numérica. Los modelos de sistemas subyacentes y las series temporales de datos que generan los procesos son generalmente complejos para estas aplicaciones y los modelos para estos sistemas generalmente no se conocen a *priori*. La estimación precisa e imparcial de las series temporales de datos producidos por estos sistemas, no siempre se puede lograr usando técnicas lineales conocidas; y por lo tanto, el proceso de estimación requiere algoritmos más avanzados de predicción de series temporales.

Para todos estos tipos de aplicaciones se busca encontrar un modelo, el cual permita predecir los futuros valores del conjunto de datos con el mayor porcentaje de precisión posible. Son numerosos los enfoques y métodos para poder resolver este problema, desde los modelos estadísticos como los ARIMA (acrónimo del inglés *AutoRegressive Integrated Moving Average*) y las funciones de transferencia (Box *et al*., 2011; Makridakis *et al*., 2008), hasta los modelos no lineales (Granger y Terasvirta, 2011). En particular, cuando el conjunto de datos presenta características no lineales, se han aplicado con resultados favorables los modelos de redes neuronales artificiales (ANN, de las voces inglesas *Artificial Neural Networks*). Zhang *et al*. (1998) nos ofrecen una visión general de este tipo de redes, mientras que hemos podido encontrar determinadas aplicaciones en (Heravi *et al*., 2004; Swanson y White, 1997; Faraway y Chatfield, 1998; Zhang, 2003; Cottrell *et al*., 1995; De Groot y Wuertz, 1991; Darbellay y Slama, 2000; Kuan y Liu, 1995).

Otro de los modelos a tener en cuenta para la predicción de series no lineales es el correspondiente a los perceptrones multicapa (MLP, de los términos ingleses *MultiLayer Perceptron*), ya que son capaces de aproximar cualquier función continua en un dominio compacto (Hornik *et al*., 1989; Cybenko, 1989). No obstante, su especificación se basa en fundamentos heurísticos y en el juicio experto del modelador (Masters, 1993, 1995). De este modo, dicho proceso se fundamenta en una serie de pasos críticos que afectan al modelo (Kaastra y Boyd, 1996), en cuanto al ajuste de datos históricos se refiere.

En cuanto a las máquinas de soporte vectorial (SVM, de los términos ingleses *Support Vector Machines*) (Vapnik, 1999; Vapnik *et al*., 1997) son una clase de red neuronal, originalmente diseñada para solucionar los problemas no lineales de

clasificación (Burges, 1998; Belousov *et al*., 2002), aunque en los últimos años se han venido aplicando a la regresión (Vapnik *et al*., 1997; Smola y Schölkopf, 2004; Lu *et al*., 2009) y a la predicción de series temporales (Mukherjee *et al*., 1997; Müller *et al*., 1997; Osowski y Garanty, 2007; Tay y Cao, 2001; Thissen *et al*., 2003; Kim, 2003; Cao, 2003; Lu *et al*., 2009; Bose y Raktim, 2005; Pai y Lin, 2005).

### 2.4.1 Enfoques sobre la predicción de series temporales

Existen trabajos (Górriz *et al*., 2004b) en los que se implementa una red neuronal paralela para la predicción de series temporales mediante PVM (acrónimo del inglés *Parallel Virtual Machine*)[1] y MPI (acrónimo de las voces inglesas *Message Passing Interface*)[2], los cuales son paquetes de software populares para la programación paralela conjunta en estaciones de trabajo, a fin de conseguir simulaciones en tiempo real. Todo ello con el objetivo de reducir el tiempo de cálculo. En la paralelización se consigue un doble objetivo: (i) la actualización de los parámetros autorregresivos utilizando un algoritmo genético y (ii) la evaluación de la función de predicción global a través de una red neuronal paralela.

En la misma línea, Górriz *et al*. (2004a) proponen un nuevo método para la predicción de series temporales volátiles usando algoritmos y filtrado para el análisis de componentes independientes (ICA, del inglés *Independent Component Analysis*) como herramientas de pre-procesamiento. Los datos pre-procesados se introducen en una red neuronal artificial basada en funciones de base radial (RBF, de las voces inglesas *Radial Basis Functions*) y la predicción resultante se compara con los resultados anteriores obtenidos sin estas herramientas de pre-procesamiento o con el alto coste computacional basado en redes neuronales paralelas (PNN, del inglés *Parallel Neural Networks*).

También existen diferentes trabajos en el campo de las series temporales, los cuales usan modelos de aprendizaje automático para poder obtener resultados concluyentes. Más concretamente, estos trabajos se centran en el modelo SVM (acrónimo de las voces inglesas *Support Vector Machine*). La motivación de Sapankevych y Sankar (2009) para usar SVM es la capacidad de esta metodología para predecir con exactitud los datos de series temporales cuando los procesos subyacentes del sistema, normalmente, son no lineales, no estacionarios y no

[1] PVM es un sistema software que permite a un conjunto de equipos heterogéneos ser usados como un sistema coherente y flexible de recursos de cálculo concurrente.

[2] MPI es una biblioteca de funciones y macros destinados a ser utilizados en los programas que se aprovechan de la existencia de múltiples procesadores mediante el paso de mensajes.

se definen a *priori*. SVM también ha sido utilizado por los autores para superar a otras técnicas no lineales, incluyendo técnicas de predicción no lineales basadas en la red neuronal como los perceptrones multicapa. El objetivo final es proporcionar una idea de las aplicaciones que utilizan SVM para la predicción de series temporales y esbozar algunas de las ventajas y desafíos que el uso de esta técnica proporciona.

Otro clásico ejemplo en este área es la predicción de resultados electorales. El resultado final es el usado para juzgar la precisión de una medida particular de predicción, en lugar de una serie continua. Tumasjan *et al*. (2010) trabajaron en las elecciones federales al parlamento nacional alemán de 2009, en donde los autores encontraron que el volumen de participación en *Twitter* refleja con exactitud la distribución de los votos en la elección entre los 6 partidos principales. También es destacable el estudio de Kim y Hovy (2007) que extraen el contenido de un sitio web de predicción política y entrenan los clasificadores con las características léxicas para identificar el *sentimiento predictivo*, con resultados prometedores para la predicción en las elecciones locales canadienses.

En cuanto al uso de las series temporales y la predicción para discernir el comportamiento de los usuarios en distintas situaciones y plataformas web, Mohan (2009) estudia la metodología en el análisis de las series temporales y las adapta para analizar los aspectos temporales en la interacción individual del usuario con los motores de búsqueda, según consta en los registros de búsqueda. Para ello, explora 2 de los enfoques más conocidos para desarrollar las ecuaciones: (i) el método de regresión y (ii) el método de ARIMA, donde cada método tiene sus propias ventajas. El autor opta por el método de regresión, ya que podía controlar los datos de los eventos discretos de un usuario individual que figuran en los registros de los motores de búsqueda, mientras que emplea ARIMA para las predicciones más a largo plazo o las predicciones basadas en el análisis de todas las actividades del usuario.

En la misma línea, Zhang *et al*. (2009) emplean el análisis de series temporales para evaluar escenarios futuros usando los registros de los motores de búsqueda, para poder desarrollar modelos para el análisis de los comportamientos de los usuarios que buscan información a lo largo del tiempo y para investigar si el análisis de las series temporales es un método válido para la predicción de las relaciones entre las acciones de los usuarios. El estudio está llevado a cabo en 2 fases. En la fase inicial, emplearon un análisis básico en el que encontraron que el 10 % de los usuarios hacían *clic* en enlaces patrocinados. En la segunda fase, utilizaron un método de análisis de series temporales con 1 salto de predicción

temporal junto con una función de transferencia[1].

Otra investigación, en referencia al comportamiento de los usuarios, es la llevada a cabo por Mayrhofer *et al*. (2003). Aunque la temática es diferente (dispositivos móviles) el concepto de modelar el comportamiento de los usuarios, tanto en el presente como en el futuro, también es aplicable. En esta investigación, los autores presentan una arquitectura que permite a los dispositivos móviles reconocer el contexto actual del usuario y anticiparse a contextos futuros del mismo. Los retos principales a los que los autores se enfrentan son:

- La integración del reconocimiento del contexto y la predicción en los dispositivos móviles, ya que disponen de recursos limitados.
- El aprendizaje y la adaptación deben ocurrir en línea, sin fases de entrenamiento explícitas.
- La intervención del usuario debe minimizarse, mientras su interacción no sea intrusiva.

Para lograr esto, la arquitectura que presentan consta de 4 partes principales: (i) la extracción de características, (ii) la clasificación, (iii) el etiquetado y (iv) la predicción.

## 2.5 Sumario

En este capítulo hemos repasado la literatura perteneciente a los diferentes enfoques y métodos, en cuanto al procesado de textos y datos, al filtrado de contenidos y textos, a la reputación y relevancia de los usuarios en diferentes sitios web y a las series temporales y predicción se refiere.

Una vez realizada una pequeña introducción al estado del arte, hemos descrito los principales métodos para poder realizar el procesamiento de datos y textos. Concretando, hemos explicado el descubrimiento de conocimiento, la minería de datos (y su relación con ciertas áreas de la investigación) y la minería de textos. A su vez, también hemos desarrollado otras áreas de investigación relevantes en este ámbito: (i) la recuperación de información, (ii) el procesamiento del lenguaje natural y (iii) la extracción de información, las cuales toman especial relevancia en el desarrollo de esta tesis.

[1]Una función de transferencia es un modelo matemático que relaciona la salida de un sistema con la señal de entrada del mismo mediante un cociente.

En segundo lugar, hemos estudiado el filtrado de contenidos y el filtrado de textos como parte perteneciente a los diferentes tipos de métodos para el propio filtrado de datos; ya que una vez procesados los datos, será necesario filtrarlos en base a ciertos parámetros. Para ello, hemos expuesto diversos enfoques para el filtrado web y la valoración. En cuanto al filtrado de textos, también hemos mostrado diferentes métodos, los cuales son perfectamente aplicables a nuestra investigación, con el fin de resolver la problemática hallada.

A continuación, hemos revisado distintos métodos de reputación y relevancia de usuarios en entornos web. Primeramente, hemos definido el concepto de reputación, para proceder a la exposición de una serie de motores computacionales de reputación y sistemas de reputación empleados por algunos de los sitios web comerciales más relevantes en el uso de esta técnica como son: (i) *eBay*, (ii) *Amazon* o (iii) *Google*.

En último lugar, hemos desarrollado el concepto de las series temporales, así como de la predicción temporal, mediante explicaciones sobre las mismas y enfoques que emplean la predicción de series temporales en entornos tan diversos como la ciencia, la política o el comportamiento de las personas.

*«El placer no está en descubrir la verdad,*
*sino en el esfuerzo de buscarla.»*

Liev Nikoláievich Tolstói (1828-1910)

CAPÍTULO

# 3

# Modelado y procesado inicial de los datos

DURANTE el desarrollo de este capítulo, explicamos los métodos y técnicas empleados para el modelado y procesado inicial de los datos descargados del sitio web social *Menéame*, con el fin de analizarlos, procesarlos y posteriormente, obtener resultados y conclusiones. Debido a que el conjunto de datos obtenido de *Menéame* es de gran envergadura, se antoja necesario un procesamiento inicial, en el que hemos obtenido la información necesaria en el formato adecuado. Para poder representar los datos de manera apropiada, hemos empleado el modelo de espacio vectorial (VSM, del inglés *Vector Space Model*), después de haber extraído correctamente la información de nuestro conjunto de datos.

Por ello, podemos diferenciar 2 etapas dentro de este proceso inicial: (i) una primera etapa de descarga de información del conjunto de datos y su etiquetado manual y (ii) una segunda etapa, en la cual se procede a la adaptación de esta información al formato requerido y necesario para que pueda ser empleada ulteriormente.

En resumen, las principales contribuciones son las siguientes:

- Un nuevo método para representar comentarios en sitios web sociales.
- Un procedimiento novel para moderar usuarios en sitios web sociales.

El resto del capítulo está organizado de la siguiente manera. La sección 3.1 explica el sitio web social elegido para la extracción del conjunto de datos a procesar. La sección 3.2 desarrolla los pasos empleados para el procesado inicial y etiquetado del conjunto de datos, así como los resultados obtenidos. La sección 3.3 hace referencia a la adaptación de los datos, mediante la extracción de características del conjunto de comentarios y de usuarios y los resultados logrados. Por último, la sección 3.4 resume los conceptos expresados a lo largo de este capítulo.

## 3.1 Sitio web social escogido: *Meneame.net*

*Menéame*[1] es un sitio web social, inspirado en el *Digg*[2] estadounidense, en el cual se promocionan diferentes tipos de noticias o historias. Desarrollado a finales de 2005 por Ricardo Galli y Benjamí Villoslada, actualmente está liberado como software libre bajo la licencia Affero GPL (*Affero General Public License*). El portal web se centró en un principio en temas científicos y tecnológicos, pero finalmente accedió a publicar contenidos sobre política, sociedad o deportes debido al gran auge de este tipo de contribuciones por parte de los usuarios.

Los usuarios son una parte muy importante de *Menéa*me. Existen 5 clases de usuarios dependiendo de la actividad de los mismos: (i) normal, (ii) *special*, (iii) *blogger*, (iv) *admin* y (v) *god*. Los diferentes usuarios de *Menéame* son valorados dependiendo del *karma* (prestigio, valoración) que posean. Este *karma* se define por un valor entre 1 y 20 (siendo 6 el valor asignado al registrarse un usuario por primera vez). El cálculo del *karma* se hace a partir de la actividad realizada por el usuario en el sistema en los 2 días anteriores a la ejecución del proceso, gracias a un algoritmo que combina 4 elementos: (i) los votos recibidos sobre las noticias, (ii) los votos positivos emitidos, (iii) los votos negativos emitidos y (iv) los votos recibidos sobre sus comentarios. Una vez definido el tipo de usuario dentro del sitio web social, éste dispone de 2 tipos de estados: (i) *disabled* y (ii) *autodisabled*. Si se abusa del sistema, el usuario tendrá un estado *disabled*, el cual deshabilita su cuenta. Si el propio usuario se ha dado de baja, adquirirá el estado *autodisabled* y no podrá loguearse, pero sí seguir visitando noticias.

El funcionamiento correcto de *Menéame* depende directamente de los usuarios, ya que son los principales responsables y encargados de interactuar con el sitio social. El funcionamiento es el siguiente: cualquier usuario de *Menéame*

[1] `http://www.meneame.net`
[2] `http://digg.com`

(registrado o no) es capaz de votar las noticias en la portada o en la sección *cola de pendientes*, pero únicamente los usuarios registrados en el sitio web serán los encargados de enviar noticias al mismo. Dichas contribuciones llegarán a la denominada *cola de pendientes*, en la cual serán votadas por los lectores. Sin embargo, solamente los usuarios registrados pueden ejercer el voto negativo y comentarlas. Los distintos *karmas* de los usuarios que votan la noticia son sumados para poder elaborar el *karma* acumulado de la misma y en el caso de que esta supere un umbral mínimo, son publicadas en la portada de *Menéame*. En cambio, las noticias que acumulen ciertos votos negativos, serán llevadas a la *cola de descartadas*. Estas contribuciones se etiquetarán como: (i) irrelevante, (ii) antigua, (iii) cansina, (iv) sensacionalista, (v) *spam*, (vi) duplicada, (vii) *microblogging*, (viii) errónea y (ix) copia/plagio. Ninguna clase de usuario puede eliminar o modificar el *karma* de las noticias o editar el voto de las mismas.

En la Figura 3.1 se puede observar el esqueleto de una noticia en *Menéame*, así como los comentarios de los usuarios.

El título expuesto será el mismo que el de la noticia en cuestión; seguido a él, se colocará el enlace que permita la redirección a la página *original*. Automáticamente, y a continuación del enlace, se genera el identificativo del usuario que envía la noticia, así como la fecha en la que la envío y la fecha en la cual se publicó. Se debe redactar una *entradilla*, que debe ser aclarativa con la noticia, para que los futuros lectores de la misma puedan llegar a saber el interés que les produce y sirva de pequeño resumen. Inmediatamente debajo, se encuentran las etiquetas. Gracias a éstas se podrá indexar de mejor manera las noticias, con lo que el buscador podrá encontrar de forma real los artículos relacionados. Continuando descendentemente, se encuentran los votos negativos, los usuarios y los anónimos[1]. En la zona más inferior de la estructura de la noticia, se puede observar el número de comentarios que posee, así como los temas involucrados (sociedad, cultura, deportes, etc.) y el *karma* de la noticia.

A continuación de la noticia, se encuentran los comentarios. Parte fundamental de *Menéame*, ya que en ellos los usuarios pueden entrar a debate con otros miembros mediante sus opiniones personales u otros tipos de comentarios. A su vez, dichos comentarios fomentan la réplica de los mismos y la producción de un mayor número de artículos, a veces relacionados con esos comentarios. En primer lugar, aparece el número de comentario perteneciente a la noticia. Seguidamente, aparece otro número (si fuese el caso) indicando que este comentario no se refiere a la noticia, sino que comenta una opinión de un usuario anterior. Dentro del comentario se puede expresar de muchas formas el pare-

[1]Los votos anónimos son los votos realizados por usuarios no registrados.


**Figura 3.1:** Ejemplo de noticia con comentarios en *Menéame*.

cer del usuario. Desde los típicos emoticonos para mostrar alguna emoción del autor, hasta el aporte de alguna fuente de datos que corrobore el texto como enlaces externos a páginas web.

## 3.2 Procesado inicial de los datos

Para la realización de esta etapa, hemos seguido una serie de pasos. Primeramente, la implementación de una herramienta para la descarga del conjunto de datos de comentarios. Una vez descargados todos los comentarios requeridos, hemos formado el conjunto de datos con todos ellos. Posteriormente, hemos etiquetado todos los comentarios manualmente mediante una aplicación desarrollada para tal labor.

En segundo lugar, y siguiendo las mismas pautas que en la etapa anterior, hemos programado una solución para proceder a la descarga de los usuarios necesarios, contenidos en los comentarios del conjunto de datos de comentarios. Al finalizar la descarga, hemos unido los usuarios con sus respectivos comentarios para formar el conjunto de datos de usuarios. A continuación, hemos procedido al etiquetado del conjunto de datos, de manera manual, gracias a la implementación de una solución que permite el etiquetado manual de los usuarios y sus comentarios.

### 3.2.1 **Proceso automatizado de adquisición de datos**

El proceso de descarga del conjunto de datos de comentarios, lo hemos realizado de manera automatizada mediante la ejecución de una aplicación desarrollada en el lenguaje de programación C#[1], como parte de la plataforma *.NET*[2]. Para ello, planificamos la descarga de la información todos los días a las 10:00 a.m. desde el 5 de Abril hasta el 12 de Abril de 2011. Esta información necesaria, la hemos obtenido al recuperar todos los comentarios de cada una de las noticias que en ese momento se encontraban en la portada del sitio web social *Menéame*.

Gracias a este proceso automático de descarga, hemos creado un conjunto de datos con un total de 9.044 comentarios, divididos en 50 noticias al día durante los 7 días que duró el proceso. Cada una de las noticias descargada es almacenada en un archivo de texto con la extensión *.processed*.

En el mismo sentido, hemos realizado la descarga de usuarios. Para llevar a cabo este conjunto de datos, hemos extraído todos los usuarios que habían realizado al menos un comentario durante la semana de descarga del conjunto de comentarios. Posteriormente, hemos descargado las características del perfil de cada usuario perteneciente al conjunto de datos de usuarios. En alguno de los casos, los usuarios estaban desactivados en el sistema de *Menéame*, pero ello no impidió su descarga ya que el propio *Menéame* guarda los perfiles de dichos usuarios durante un cierto periodo de tiempo. En el caso de no existir algunos usuarios, hemos obviado sus comentarios a la hora de realizar este conjunto de datos.

Por lo tanto, el conjunto de datos de usuarios, lo forman las características del perfil de cada usuario y la actividad del mismo, reflejada en los comentarios

[1] C# es un lenguaje de programación orientado a objetos desarrollado por *Microsoft* en el año 2000.

[2] La plataforma *.NET* es un entorno de trabajo *software* lanzado por *Microsoft* en 2002, que permite un desarrollo rápido para la programación de aplicaciones.

que realizó dicho usuario durante las fechas del proceso de descarga automática del conjunto de comentarios. Gracias a ello, hemos formado un conjunto de datos de 3.359 usuarios.

### 3.2.2 Descripción del conjunto de datos

En esta sección, explicamos ambos conjuntos de datos, tanto el conjunto de datos de comentarios como el perteneciente a los usuarios, para tener una visión más clara y sencilla sobre el tipo de datos recopilados.

#### 3.2.2.1 Conjunto de comentarios

En este apartado, explicamos el conjunto de datos de los comentarios mediante un ejemplo de noticia y otro de comentario, con el fin de entender mejor las características extraídas, que finalmente formarán el archivo a procesar. Una explicación más detallada sobre la noticia, el comentario y sus características se ofrece en la sección 3.1.

La Figura 3.2 muestra la estructura de una noticia cuando se encuentra en la portada. El título de la noticia debe ser el mismo que el de la noticia externa. Después del título, el usuario enlaza la noticia. También se debe escribir un breve resumen de la noticia que resulte descriptivo para los usuarios que lo lean. En el pie de la noticia, se encuentran las etiquetas, así como los diferentes tipos de votos. Debajo de esta información, se encuentra el número de comentarios, junto con las categorías que *Menéame* aplica y el valor del *karma*. Además, en la parte izquierda, se muestran el número de votos positivos de la noticia.


**Figura 3.2:** Estructura de una noticia en *Menéame*.

Por otra parte, la Figura 3.3 muestra un comentario de *Menéame*. El primer símbolo que aparece es el número de comentario, en este caso el 7. Seguido,

se aprecia otro número que referencia a otro comentario. En este caso, el usuario está dando su opinión sobre un comentario previo, el 4º comentario de la noticia. A continuación, se expresa la opinión del usuario y justo debajo de la misma aparecen diferentes término referidos al comentario como: (i) votos, (ii) *karma*, (iii) fecha de publicación y (iv) usuario. En todo caso, todos estos valores se refieren siempre al comentario, por lo que el valor del *karma* del comentario indica la suma de los *karmas* de los usuarios que han votado ese comentario, junto con el *karma* del propio usuario.

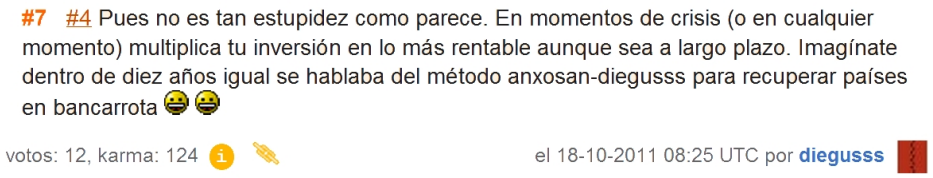



**Figura 3.3:** Estructura de un comentario en *Menéame*.

La estructura de los archivos, en los cuales se encuentran las noticias y sus respectivos comentarios descargados, es la siguiente:

- **Título.** Es el título de la noticia (el mismo que el de la historia externa al que hace referencia).
- **Resumen.** Es una pequeña descripción de la noticia, la cual es identificativa y aclarativa con la noticia.
- **Etiquetas.** Son las palabras clave de la noticia relacionadas con el contenido del enlace, no con la temática de la misma. El número aproximado de etiquetas es de 4 o 5 palabras y las define el usuario.

A continuación de las etiquetas, se exponen ordenados numéricamente todos los comentarios de los que consta la noticia en el momento de la descarga; ya que dicha noticia puede disponer de un mayor número de comentarios en los días posteriores. Estos comentarios están formados por diferentes elementos:

- **Número de comentario.** Es el número que indica la posición del comentario publicado por el usuario, asignados de menor a mayor según el orden de publicación. Su formato es el compuesto por el carácter # seguido del número correspondiente.

- **Número del comentario respondido.** En el caso de que un comentario realizado por un usuario no haga referencia a la noticia, si no a otro comentario, éste número será el correspondiente a tal comentario. En el caso de que no exista tal réplica, no aparecerá.

- **Texto del comentario.** Lugar indicado para que el usuario exprese su opinión o parecer respecto a un tema o a un comentario escrito anteriormente por otro usuario. Se puede escribir texto, los típicos emoticonos para mostrar alguna emoción o el aporte de fuentes externas: vídeos, fotografías, enlaces externos, etc.

- **Votos.** Es el número de votos que ha recibido el comentario por parte del resto de los usuarios del sistema.

- ***Karma*.** Es la suma de los diferentes *karmas* de las personas que han votado positivamente el comentario, más el *karma* del propio usuario. En caso de no tener ningún voto, el *karma* únicamente será el perteneciente al usuario.

- **Fecha.** Es el valor que indica hace cuanto tiempo el usuario de *Menéame* publicó el comentario.

Seguidamente, la Figura 3.4 muestra el archivo de una noticia de *Menéame* con su estructura al completo, anteriormente explicada.

#### 3.2.2.2 Conjunto de usuarios

Para formar el conjunto de datos de usuarios es necesario obtener 2 fuentes de datos diferentes. Por una parte, el conjunto de datos de comentarios ya implementado y explicado en la sección 3.2.2.1, así como el conjunto de datos del perfil de los usuarios de *Menéame*, cuyo ejemplo se muestra en la Figura 3.5.

A continuación, explicamos la estructura del fichero formado. En este caso, el nombre del fichero será el alias del usuario con la extensión *.processed*:

- **Usuario.** Es el alias que identifica a la persona cuando publica comentarios, envía noticias, etc.

- **Nombre.** Es el nombre *personal* del usuario. Es un campo opcional en el perfil.

TITULO: El paro puede multiplicar por siete el riesgo de contraer enfermedades mentales.
RESUMEN: La última gran crisis del capitalismo, generada y aprovechada por banqueros, grandes empresarios y gobiernos conservadores, ha aumentado la desigualdad, la pobreza y el desempleo en el mundo, una gran parte de cuya población está desprotegida.
ETIQUETAS: paro, desempleo, enfermedad, sociedad, mentales

#1 ¿Sólo lo multiplica por siete? Realmente estoy viendo gente muy desesperada y si yo me viese en alguna de esas situaciones.....es que no sé ni lo que podría llegar a hacer, la verdad. votos: 17, karma: 146 hace 1 día 1 hora 45 minutos por murciana_cabreada

#2 Y el empleo puede multiplicar por veinte el riesgo de contraer enfermedades mentales.
www.youtube.com/watch?v=bCMXuQKisyE
Como diría Rubianes: el trabajo te pule, te abrillanta, te da esplendor... ¡hasta te pone cachondo! votos: 5, karma: 18 hace 1 día 1 hora 42 minutos por BurnTheMoney

**Figura 3.4:** Formato de archivo con una noticia de *Menéame*.


**Figura 3.5:** Estructura del perfil de un usuario en *Menéame*.

- **Bio.** Es una pequeña descripción acerca del usuario. Es un campo opcional en el perfil.
- **IM/email.** Es el email perteneciente al usuario. Es un campo privado en el perfil.
- **Sitio web.** Formado por diversa información y enlaces a sitios web sociales o *blogs* que el usuario quiera añadir a la información de su perfil. Es un campo opcional en el perfil.
- **Desde.** Es la fecha en la que el usuario se registró en el sistema.
- ***Karma*.** Es un número entre 1 y 20 que mide y evalúa la participación del usuario en el sitio.
- ***Ranking*.** Es la posición del usuario en *Menéame*, en términos de *karma*.
- **Noticias enviadas.** Es el número de noticias compartidas, que el usuario ha enviado a *Menéame*, teniendo en cuenta las noticias publicadas y las no publicadas.
- **Entropía.** Es el coeficiente que mide la diversidad del usuario, en cuanto a la distribución de sitios como fuentes de sus noticias. Si todas las noticias enviadas parten de la misma fuente, su entropía será de valor 0. Mientras que tendrá el valor 1 en caso de que las noticias sean de enlaces diferentes.
- **Noticias publicadas.** Es el número de noticias compartidas publicadas en la portada de *Menéame*.
- **Comentarios.** Es el número total de comentarios que el usuario ha realizado desde su registro de entrada.
- **Notas.** Es el número de notas publicadas en el sitio *Nótame*[1].
- **Número de votos.** Indica los votos hechos por el usuario durante los últimos meses.
- **Geolocalización.** Es la sección en donde el usuario puede indicar la información sobre su localización física.
- ***Avatar*.** Es la imagen seleccionada por el usuario para representarle. Es un campo opcional en el perfil.

[1] `http://www.meneame.net/notame`

Una vez enumeradas todas las características del perfil de los usuarios, en el archivo de cada usuario únicamente aparecerán ciertas características (explicadas en la sección 3.3.2), ya que el resto de ellas no aportan información válida para el proceso de moderación. Debajo de estas características extraídas del perfil del usuario, se encuentran todos los comentarios etiquetados del conjunto de datos de comentarios del usuario en cuestión.

Seguidamente, la Figura 3.6 expone el archivo de un usuario de *Menéame* con su estructura al completo.

```
usuario:ThePolman666
desde:14-10-2010 15:57
karma:6.10
ranking:13874
noticias enviadas:0
noticias publicadas:0
comentarios:213
notas:0
número de votos:117
avatar:rock into mordor

#5 #3 go to #2 votos: 18, karma: 149 hace 20 horas 3 minutos
por ThePolman666
C:Irrelevante-Comentario-Normal

#25 O RLY? (facepalm) votos: 1, karma: 15 hace 11 horas 52
minutos por ThePolman666
C:Irrelevante-Noticia-Gracioso
```

**Figura 3.6:** Formato de archivo con usuario de *Menéame*.

### 3.2.3 Proceso de etiquetado

#### 3.2.3.1 Conjunto de comentarios

Hemos categorizado los comentarios en 3 clasificaciones diferentes. Con el fin de hacer la explicación más clara, mostramos ejemplos actuales de las diferentes categorías para cada una de las distintas clasificaciones.

Cada una de las 3 diferentes clasificaciones tiene varias clases posibles. Las hemos explicado a continuación:

- **Tipo de comentario.** El tipo de comentario indica qué clase de información y de qué tipo realiza el usuario. A su vez se divide en:

  - *Aporte.* El usuario contribuye añadiendo nueva información. Por ejemplo, la Figura 3.7.

> #201 #191 Ah, por cierto se me olvidó comentarte como detalle, en ese pueblo que te comento donde la parroquia instalo una megafonia que difunde rezos la mañana del domingo antes de la misa por todo el pueblo haciendo de despertador si quieres como si no, esa megafonia se instaló en el pueblo con un alcalde del PSOE que no ha hecho nada a lo largo de sus mandatos por remediarlo. Resulta llamativo en contraposición a las posturas públicas que toma su partido, pero como les gusta a algunos el sillón ¿eh? votos: 0, karma: 8 hace 18 horas 23 minutos por strel

**Figura 3.7:** Ejemplo de comentario tipo aporte.

  - *Irrelevante.* Estos comentarios no contribuyen al artículo principal ni a otros comentarios anteriores. Por ejemplo, la Figura 3.8.

> #225 Suerte compañeros!!! votos: 0, karma: 6 hace 12 horas 56 minutos por pavlenka

**Figura 3.8:** Ejemplo de comentario irrelevante.

  - *Opinión.* Estos comentarios expresan la opinión particular del usuario sobre un tema tratado en la noticia o en otros comentarios. Ejemplo en la Figura 3.9.

- **Enfoque del comentario.** El comentario puede estar enfocado tanto hacia la noticia como hacia otro comentario. Está formado por 2 opciones diferentes:

  - *Noticia.* El comentario es sobre la noticia. Por ejemplo, el expuesto en la Figura 3.10.

#205 #70 Estoy de acuerdo contigo, y es más, incluso creo que deberían hacerlo con más fuerza, porque mucho nos ha costado liberarnos; de una creencia o al menos reducirla (y digo el liberarnos muy matizado) como para dejar que nos vengan a tocar la moral con otra, y que además por razones socioculturales en principio se expandirá en un futuro (rápido crecimiento demográfico de la población que profesa tal creencia) votos: 1, karma: -1 hace 18 horas 1 minuto por RaistlinMajere

**Figura 3.9:** Ejemplo de comentario de opinión.

#202 Lo de siempre: uno se ofende porque quiere, no porque debe. Si tuvieran el punto de mira apuntando más allá se darían cuenta de que hay más cosas a parte de ellos y su verdad pero ese maldito egocentrismo y victimismo de que todo está contra mí... votos: 0, karma: 6 hace 18 horas 13 minutos por Natxo-Pistatxo

**Figura 3.10:** Ejemplo de comentario enfocado a la noticia.

#204 #201 Entonces ¿no se puede ser del PSOE y creyente?.... Es más, teniendo en cuenta que el PSOE es realmente un partido de centro derecha, no veo donde está lo que te resulta tan llamativo. ¿lo entenderias si fuese del PP?. Lo digo porque para muchos de nosotros PSOE=PP ..... votos: 0, karma: 6 hace 18 horas 8 minutos * por xaphoo

**Figura 3.11:** Ejemplo de comentario enfocado a otro comentario.

  - *Comentario.* El comentario se refiere a otro comentario de la noticia. Por ejemplo, la Figura 3.11.

- **Grado del comentario.** Hemos categorizado el nivel de controversia en 4 grados diferentes: normal, polémico, muy polémico y, también, un atributo adicional que se utiliza para comentarios graciosos o irónicos.

  - *Normal.* Un comentario normal es aquel que no plantea ninguna controversia o ironía. No busca hacer daño ni ser hiriente. Ejemplo en la Figura 3.12.

> #205 #70 Estoy de acuerdo contigo, y es más, incluso creo que deberían hacerlo con más fuerza, porque mucho nos ha costado liberarnos; de una creencia o al menos reducirla (y digo el liberarnos muy matizado) como para dejar que nos vengan a tocar la moral con otra, y que además por razones socioculturales en principio se expandirá en un futuro (rápido crecimiento demográfico de la población que profesa tal creencia) votos: 1, karma: -1 hace 18 horas 1 minuto por RaistlinMajere

**Figura 3.12:** Ejemplo de comentario normal.

– *Polémico.* Un comentario que, a propósito, busca la polémica con un tono perjudicial pero no de una manera exagerada. Por ejemplo, el comentario de la Figura 3.13.

> #218 muchos golpes de pecho y mucho ateismo y el 90% de todos ellos le ponen juguetes a sus hijos el día de reyes...ah, y vuelvan a freirme a negativos si procede, que me importa un soberano carajo. votos: 0, karma: 6 hace 14 horas 46 minutos * por canaam

**Figura 3.13:** Ejemplo de comentario polémico.

– *MuyPolémico.* Busca crear controversia en un modo exagerado, siendo perjudicial o irrespetuoso. Es lo conocido como usuario trol, como se muestra en la Figura 3.14.

> #206 #10 ¿Que no ha habido protestas contra el uso del burka en España? Y no solo de ateos, por supuesto. El día que haya el mismo número de gilipollas integristas cristianos y musulmanes en España, seremos los primeros en proclamar vuestro gilipollismo a los cuatro vientos. No os tenemos miedo ni a vosotros, ni a ellos. votos: 0, karma: 6 hace 17 horas 57 minutos por Despero

**Figura 3.14:** Ejemplo de comentario muy polémico.

– *Gracioso.* Estos comentarios intentan ser graciosos o tratan de hacer alguna broma, incluso irónicamente. Se expone un ejemplo en la Figura 3.15.

> #221 Si mis colegas y yo nos fuesemos una tarde a recorrer el pueblo repartiendo por las calles aceite, grasa, cera o cualquier otro producto susceptible de provocar resbalones, seguro que teniamos problemas. Lo menos nos llamarian vandalos o nos multarian, o las dos cosas. votos: 0, karma: 6 hace 13 horas 54 minutos por pacorron

**Figura 3.15:** Ejemplo de comentario gracioso.

Una vez establecidas las clases, se procede al etiquetado manual. Para ello, mediante la aplicación desarrollada, hemos etiquetado los comentarios del siguiente modo, como se presenta en la Figura 3.16.

> #204 #201 Entonces ¿no se puede ser del PSOE y creyente?.... Es más, teniendo en cuenta que el PSOE es realmente un partido de centro derecha, no veo donde está lo que te resulta tan llamativo. ¿lo entenderias si fuese del PP?. Lo digo porque para muchos de nosotros PSOE=PP ..... votos: 0, karma: 6 hace 18 horas 8 minutos * por xaphoo
>
> C: Opinion-Comentario-Normal

**Figura 3.16:** Ejemplo de comentario etiquetado.

Se observa que, primeramente procede el comentario y debajo del mismo la categorización. Mediante los caracteres *C:* se indica que comienza la tupla, en donde hemos expuesto el criterio a la hora de etiquetar los comentarios, seguido por las clases en un lugar determinado. El formato de etiquetado es el siguiente:

*C: TipoComentario-EnfoqueComentario-GradoComentario*

#### 3.2.3.2 **Conjunto de usuarios**

Para categorizar a los usuarios, hemos enfocado la tarea hacia 2 diferentes tipos de clasificación: (i) una dependiente del comportamiento global y (ii) otra respecto a su nivel de controversia. Para este fin, hemos recuperado varias características que se describen en la sección 3.2.2.2.

- **Tipo de Usuario.** Respecto al comportamiento de los usuarios, nos hemos centrado en el nivel de participación del usuario y en el enfoque de su actividad. De esta manera, hemos establecido las siguientes categorías:

– *Contribuyente.* Un usuario lo hemos etiquetado como contribuyente, cuando su número de comentarios respecto a su fecha de registro es elevada (más de 1 comentario al día) y a su vez, envía noticias con asiduidad para que sean publicadas. Normalmente, sus comentarios son aportes a la noticias o a otros comentarios.


**Figura 3.17:** Usuario contribuyente.

La Figura 3.17 muestra un ejemplo de un usuario contribuyente. En este caso, el usuario ha publicado una media de 5 comentarios al día y han sido publicadas un 12 % del total de las noticias enviadas.

– *Participativo.* Un usuario participativo es el que comenta otras noticias o comentarios con un grado medio o alto, enviando ciertas noticias que posteriormente son publicadas. De este modo, un usuario participativo es similar a un usuario contribuyente en menor grado, pero con más comentarios que noticias publicadas.


**Figura 3.18:** Usuario participativo.

Por ejemplo, el usuario de la Figura 3.18 ha realizado 2 comentarios por día y ha enviado 101 noticias, de las cuales únicamente se han publicado 9 de ellas.

– *Comentarista*. Normalmente, un usuario comentarista realiza un gran número de comentarios, pero publica o envía pocas noticias o ninguna. Un gran número de usuarios en *Menéame* pertenecen a esta categoría.

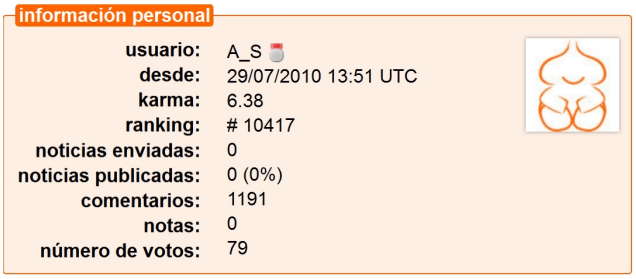



**Figura 3.19:** Usuario comentarista.

Por ejemplo, el usuario que se expone en la Figura 3.19 ha publicado 2 comentarios por día y no ha enviado ninguna noticia.

– *Lector*. Finalmente, un usuario lector se dedica a leer otros comentarios de noticias publicados, pero rara vez los comenta o comparte nuevos enlaces a noticias de páginas web.


**Figura 3.20:** Usuario lector.

Hemos considerado al usuario mostrado en la Figura 3.20 como lector; debido a que, aunque su fecha de registro fue hace bastantes tiempo, solo ha enviado 1 noticia (la cual fue publicada) y únicamente ha publicado 530 comentarios, lo que significa un ratio de comentarios con una media inferior a 1 comentario al día.

- **Grado de Usuario.** Respecto al nivel de controversia, se han desarrollado las siguientes categorías:

  - *Normal.* Un usuario normal, es aquel cuyos comentarios y noticias, en general, no suscitan controversia ni ironía.
  - *Polémico.* Hemos definido un usuario polémico como aquel cuyos comentarios y noticias buscan polémica con un tono perjudicial, a propósito.
  - *MuyPolémico.* Un usuario muy polémico es aquel cuyos comentarios y noticias buscan crear polémica de un modo exagerado, siendo perjudicial o irrespetuoso. En otras palabras, un usuario trol.
  - *Gracioso.* Este tipo de usuario suele bromear, haciendo comentarios graciosos e irónicos.

Una vez establecidas las clases, hemos procedido al etiquetado manual. Para ello, mediante la aplicación programada para tal uso, hemos etiquetado a los usuarios del siguiente modo, como muestra la Figura 3.21.

Hemos observado que en primer lugar, procede el perfil del usuario, debajo los comentarios del mismo usuario y posteriormente, un resumen de la clase de los comentarios. Por último, mediante los caracteres *Class:*, hemos indicado que comienza la tupla en donde hemos expuesto nuestro criterio a la hora de etiquetar los usuarios, seguido por las clases en un lugar determinado. El formato de etiquetado es el siguiente:

*Class: TipoUsuario-GradoUsuario*

### 3.2.4 Discusión de los resultados

Una vez etiquetados tanto los 9.044 comentarios como los 3.359 usuarios, hemos esgrimido una serie de resultados y conclusiones al respecto.

En cuanto a la primera categorización, la referente a los comentarios, observamos en la siguiente Tabla 3.1 cómo la mayoría de los comentarios se enfocan en la noticia. Sin embargo, esta clase tampoco está tan desequilibrada respecto a la otra.

La siguiente Tabla 3.2 muestra la distribución de los comentarios dependiendo de su tipo de información. La mayoría de los comentarios son irrelevantes, mientras que solamente 217 comentarios son aportes a la información contenida en el artículo.

usuario:.hF
sitio web:http://twitter.com/#!/puntohacheEFE
desde:09-03-2006 16:25
karma:10.55
ranking:734
noticias enviadas:684
entropía:43
noticias publicadas:88
comentarios:22209
notas:2455
número de votos:74
avatar:

#115 #44 Eso es como decir que la milla no es una unidad correcta de longitud porque es el kilómetro. El kilopondio (o kilogramo fuerza) es una unidad de fuerza igualmente válida (aunque no perteneciente al SI). Dimensionalmente es masa*velocidad/segundo/segundo, igual que el N. votos: 2, karma: 30 hace 13 horas 21 minutos * por .hF
C: Opinion-Comentario-Normal

#26 #25 Las dos cosas. Anteriormente le pusieron multas por exceso de velocidad y ahora le han puesto otra por pegar un frenazo. votos: 0, karma: 11 hace 13 horas 40 minutos por .hF
C: Opinion-Comentario-Normal

Comentarios Normales:2
Comentarios Polemicos:0
Comentarios Muypolemicos:0
Comentarios Graciosos:0
Comentarios Aporte:0
Comentarios Opinion:2
Comentarios Irrelevante:0
Comentarios Noticia:0
Comentarios Comentario:2
Comentarios Totales:2

Class: Contribuyente-Normal

**Figura 3.21:** Ejemplo de usuario etiquetado.

Por último, la Tabla 3.3 muestra el número de comentarios en las clases respecto al nivel de controversia del comentario, en la cual se puede observar que la mayoría de los comentarios realizados por el usuario son normales, mientras que la polemicidad (*Polémico* y *MuyPolémico*) en los comentarios no es tan alta como se había planteado en un principio, siendo únicamente el 25 % del total.

**Tabla 3.1:** Distribución de ejemplos para la categorización sobre el enfoque del comentario.

| Clase | Número de comentarios |
|---|---|
| Enfoque a la noticia | 5.191 |
| Enfoque al comentario | 3.853 |
| Total | 9.044 |

**Tabla 3.2:** Distribución de ejemplos para la categorización acerca del tipo de comentario.

| Clase | Número de comentarios |
|---|---|
| Aporte | 217 |
| Opinión | 2.460 |
| Irrelevante | 6.367 |
| Total | 9.044 |

**Tabla 3.3:** Distribución de ejemplos para la categorización acerca del grado de comentario.

| Clase | Número de comentarios |
|---|---|
| Normal | 6.327 |
| Polémico | 1.124 |
| MuyPolémico | 1.063 |
| Gracioso | 530 |
| Total | 9.044 |

En cuanto a la segunda clasificación, respecto a los usuarios, observamos en la Tabla 3.4 como una gran cantidad de los usuarios etiquetados responden al perfil de lectores. Lo cual expresa que la mayor parte de la actividad de los usuarios recogidos perteneciente al sistema no aporta demasiado, ya que durante un largo periodo pasan el tiempo leyendo las diferentes noticias y comentarios de *Menéame*.

Por otra parte, también podemos observar en la Tabla 3.5 como el comportamiento de la mayoría de los usuarios respecto a sus comentarios es normal, aunque hemos intuido un leve crecimiento de la polemicidad. El 40 % de usuarios etiquetados son polémicos o muy polémicos.

**Tabla 3.4:** Distribución de ejemplos para la categorización acerca del tipo de usuario.

| Clase | Número de comentarios |
|---|---|
| Lector | 1.671 |
| Comentarista | 1.047 |
| Participativo | 466 |
| Contribuyente | 175 |
| Total | 3.359 |

**Tabla 3.5:** Distribución de ejemplos para la categorización sobre el nivel de controversia del usuario.

| Clase | Número de comentarios |
|---|---|
| Normal | 1.604 |
| Polémico | 938 |
| MuyPolémico | 424 |
| Gracioso | 393 |
| Total | 3.359 |

## 3.3 Adaptación de los datos

En esta sección, hemos detallado las características que hemos extraído tanto del conjunto de datos de comentarios como del conjunto de datos de los usuarios.

### 3.3.1 Características extraídas de los comentarios

En el caso de los comentarios, hemos dividido estas características en 3 categorías diferentes: (i) propiedades estadísticas, (ii) propiedades sintácticas y (iii) propiedades de opinión.

#### 3.3.1.1 Propiedades estadísticas

La categoría de propiedades estadísticas dispone de varias características, las cuales hemos empleado para la construcción del conjunto final de datos.

- **Cuerpo del comentario.** Hemos utilizado la información contenida en el cuerpo del comentario. A fin de representar los comentarios, hemos

empleado el modelo de recuperación de información (IR, del inglés *Information Retrieval*). Un modelo IR puede ser definido como una 4-tupla de la manera $[\mathscr{C}, \mathscr{Q}, \mathscr{F}, \mathscr{R}(q_i, c_j)]$ (Baeza-Yates y Ribeiro-Neto, 1999) donde:

- $\mathscr{C}$ es el conjunto de representaciones de comentarios.
- $\mathscr{Q}$ es el conjunto de consultas del usuario.
- $\mathscr{F}$ es el marco de trabajo para modelar comentarios, consultas y sus relaciones.
- $\mathscr{R}(q_i, c_j)$ es una función de clasificación que asocia un número real con una consulta $q_i$ ($q_i \in \mathscr{Q}$) y la representación de un comentario $c_j$, por lo que ($c_j \in \mathscr{C}$).

Como $\mathscr{C}$ es el conjunto de comentarios $c$, $\{c : \{t_1, t_2, ...t_n\}\}$, donde cada uno de los comentarios comprende $n$ términos $t_1, t_2, \ldots, t_n$, hemos definido el peso $w_{i,j}$ como el número de veces que aparece el término $t_i$ en el comentario $c_j$. Si $w_{i,j}$ no está presente en $c$, el peso $w_{i,j} = 0$. Por lo tanto, un comentario $c_j$ puede ser representado como un vector de pesos $\vec{c_j} = (w_{1,j}, w_{2,j}, ...w_{n,j})$.

Sobre la base de esta formalización, los sistemas IR normalmente emplean el modelo espacio vectorial (VSM, de los términos ingleses *Vector Space Model*) (Baeza-Yates y Ribeiro-Neto, 1999), el cual representa algrebaicamente documentos (en nuestro caso comentarios) como vectores en un espacio multi-dimensional. Este espacio se compone, únicamente, de las intersecciones de los ejes positivos. Los comentarios son representados por una matriz *término-por-comentario*, donde el elemento $(i, j)$ ilustra la asociación entre el término $i$ y el comentario $j$. Esta asociación refleja la aparición del término $i$ en el comentario $j$. Los términos pueden representar diferentes unidades textuales (por ejemplo, palabras o frases) y también pueden ser ponderados individualmente, permitiendo a los términos convertirse en más o menos importantes dentro de un comentario dado o dentro de una colección de comentarios $\mathscr{C}$ al completo.

También hemos utilizado el esquema de ponderación denominado frecuencia del término-frecuencia inversa del documento (TF-IDF, de su denominación inglesa *Term Frequency-Inverse Document Frequency*) (Salton y McGill, 1983), donde el peso del término $i$ en el comentario $j$, denotado por $weight(i, j)$, es definido por:

$$weight(i, j) = tf_{i,j} \cdot idf_i \tag{3.1}$$

donde la frecuencia del término $tf_{i,j}$ es definida como:

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{3.2}$$

donde $n_{i,j}$ es el número de veces que el término $t_{i,j}$ aparece en el comentario $c$ y $\sum_k n_{k,j}$ es el número total de términos en el comentario $c$. Por otra parte, la frecuencia inversa del documento, se define como:

$$idf_i = \frac{|\mathscr{C}|}{|\mathscr{C} : t_i \in c|} \tag{3.3}$$

donde $|\mathscr{C}|$ es el número total de comentarios y $|\mathscr{C} : t_i \in c|$ es el número de comentarios que contienen el término $t_i$.

Una vez denominado el esquema, hemos empleado 2 alternativas diferentes. Primero, hemos empleado la palabra como término. Y segundo, hemos utilizado el enfoque *n-grama* como término, con el valor máximo de $n = 3$ y el valor mínimo de $n = 1$. El enfoque *n-grama* nos ofrece la posibilidad de crear una subsecuencia de $n$ elementos consecutivos en una secuencia dada, en este caso en un comentario dado.

Con el fin de construir el modelo, también hemos suprimido las palabras de parada (del inglés *stop-words*)(Wilbur y Sirotkin, 1992), las cuales son palabras desprovistas de contenido o palabras las cuales su aparición en el idioma empleado es muy habitual, con lo cual no aportan información alguna (por ejemplo, un, el, es). Estas palabras no proporcionan ninguna información semántica y añaden ruido al modelo (Salton y McGill, 1983). Hemos utilizado una lista de palabras de parada externa de términos en castellano[1].

- **Número de referencias al comentario.** Nos indica el número de veces que el comentario ha sido referenciado en otros comentarios de la misma historia. En *Menéame*, esta referencia se indica por el símbolo # seguido por el número del comentario. Esta medida es importante y efectiva a la hora de capturar la importancia de un comentario en una discusión.

- **Número de referencias desde el comentario.** Esta medida indica el número de referencias en un comentario a otros comentarios de la misma

[1]La lista de palabras de parada puede descargarse en: `http://paginaspersonales.deusto.es/isantos/resources/stopwords.txt`

historia. Hemos considerado que esta característica es capaz de capturar si el comentario está hablando sobre la noticia o, en cambio, sobre otro comentario.

- **Número de comentario.** También hemos recuperado el número del comentario, el cual nos indica la antigüedad del mismo. En *Menéame*, al igual que ocurre en otros sitios web sociales y foros, si una noticia tiene un alto número de comentarios, el tema suele derivar a una discusión que difiere de la noticia, en ocasiones, de manera polémica.

- **Similitud del comentario con el resumen de la noticia.** Hemos empleado la similitud del VSM del comentario con el modelo del resumen de la noticia. En particular, hemos usado la similitud del coseno (Tata y Patel, 2007):

$$sim(\vec{v},\vec{u}) = \cos(\theta) = \frac{\vec{v}\cdot\vec{u}}{||\vec{v}||\cdot||\vec{u}||} \tag{3.4}$$

  donde $\vec{v}\cdot\vec{u}$ es el producto escalar de $\vec{v}$ y $\vec{u}$, mientras que $||\vec{v}||\cdot||\vec{u}||$ es el producto de los módulos de $\vec{v}$ y $\vec{u}$.

  Este valor varía entre 0 y 1, donde el valor 0 significa que los 2 documentos son completamente diferentes; es decir, los vectores son ortogonales entre ellos. Por contra, si los 2 documentos son similares entre sí, el ángulo que forman es pequeño, con lo que la similitud del coseno tiende al valor 1.

  Hemos utilizado esta característica debido a que puede indicarnos cuánto se parece el comentario a la noticia.

- **Número de coincidencias entre las palabras del comentario y las etiquetas de la noticia.** Hemos contado el número de palabras que aparecen en el comentario y que aparecen, a su vez, como etiquetas de la historia. Hemos usado esta medida, ya que puede indicarnos cómo de relacionado está el comentario respecto a la noticia.

- **Número de enlaces en el cuerpo del comentario.** También hemos contabilizado el número de enlaces dentro del cuerpo del comentario. Esta medida intenta indicarnos si un comentario emplea fuentes externas, con el fin de apoyar su afirmación, aunque también puede ser un enlace a un contenido inapropiado o gracioso.

#### 3.3.1.2 **Propiedades sintácticas**

En esta categoría, hemos contabilizado el número de palabras en las diferentes categorías sintácticas. Para este fin, hemos realizado un etiquetado de POS (acrónimo del inglés *Part of Speech*) usando la herramienta *Freeling*[1]. Las siguientes características han sido utilizadas:

- Número de adjetivos en el cuerpo del comentario.
- Número de números en el cuerpo del comentario.
- Número de fechas en el cuerpo del comentario.
- Número de adverbios en el cuerpo del comentario.
- Número de nombres en el cuerpo del comentario.
- Número de conjunciones en el cuerpo del comentario.
- Número de pronombres en el cuerpo del comentario.
- Número de signos de puntuación en el cuerpo del comentario.
- Número de interjecciones en el cuerpo del comentario.
- Número de determinantes en el cuerpo del comentario.
- Número de abreviaturas en el cuerpo del comentario.
- Número de preposiciones en el cuerpo del comentario.
- Número de verbos en el cuerpo del comentario.

Estas características están destinadas a capturar el tipo de lenguaje del usuario en un comentario. Por ejemplo, un elevado uso de adjetivos suele indicar que el usuario está expresando su opinión. Mediante la captura del tipo de lenguaje, el método puede ser capaz de identificar el grado de polemicidad de un comentario, así como el tipo de información contenida en el comentario.

[1] Disponible en `http://www.lsi.upc.edu/~nlp/freeling`

#### 3.3.1.3 **Propiedades de opinión**

Esta categoría se refiere a las características que indican opinión. Específicamente, hemos usado las siguientes características:

- **Número de palabras positivas y negativas.** Hemos tenido en cuenta el número de palabras en el comentario con significado positivo y el número de palabras, en el mismo comentario, con significado negativo. Hemos empleado un léxico de opinión externo[1]. Debido a que los términos en este léxico están escritos en Inglés y *Menéame* está escrito en castellano, hemos traducido este léxico al español.
- **Número de votos.** Hemos utilizado el número de votos positivos del comentario. En *Menéame* los votos son otorgados por otros usuarios.
- ***Karma*.** El *karma* es calculado por el sitio web y representa cómo de importante es el comentario, basándose en la cantidad de votos positivos y negativos otorgados a ese comentario.

En este sentido, hemos usado 2 características que son externas a *Menéame*: (i) el número de palabras negativas y positivas y (ii) características que expresan opinión, que *Menéame* ya ha procesado. Las últimas son el número de votos positivos que cada comentario y el *karma*, el cual es un concepto usado en *Menéame* para moderar los comentarios y usuarios. Estas características están dedicadas a categorizar el comentario en su grado de controversia, debido a que nos indican la opinión de la comunidad *Menéame* sobre el comentario y también, la polaridad del comentario por medio de las palabras positivas o negativas.

### 3.3.2 Características extraídas de los usuarios

En el caso de los usuarios, hemos diferenciado las características en 2 categorías distintas: (i) las referentes al perfil del usuario y (ii) las relacionadas con los comentarios.

#### 3.3.2.1 **Propiedades extraídas del perfil**

Son varias las características que hemos reunido de los perfiles públicos de los usuarios para procesarlas posteriormente. En particular, las siguientes:

[1]Disponible en: `http://www.cs.uic.edu/~liub/FBS/opinion-lexicon-English.rar`

- **Fecha de registro.** Es el valor que indica cuándo el usuario se registró en el sistema.

- ***Karma*.** Es el *karma* del usuario, calculado por *Menéame* basándose en los votos de otros usuario.

- ***Ranking*.** Es la posición del usuario basada en su cantidad de *karma*.

- **Número de noticias enviadas.** Establece si el usuario proporciona enlaces a noticias o no y el número de noticias que manda.

- **Número de noticias publicadas.** Esta característica establece el número de noticias que han sido enviadas por el usuario y publicadas en la portada del sitio web social.

- **Entropía.** Calculada por *Menéame*, decreta que un usuario tiene elevada entropía si envía noticias de diferentes fuentes. Esta medida es usada por el sitio web para detectar posibles usuarios que envían *spam*, ya que se centran en ciertas fuentes o únicamente en una, con lo cual tendrán una baja entropía al no tener diversidad en sus fuentes.

- **Número de comentarios.** Indica el número de comentarios publicados por el usuario en las noticias publicadas en *Menéame*.

- **Número de notas.** Mide el número de anotaciones realizadas por el usuario en el sistema de notas relacionado con *Menéame*: *Nótame*[1], el cual no está vinculado con las noticias, por lo que puede ser visto como un foro dentro del sitio web social.

- **Número de votos.** Es el número de votos recibidos por el usuario de otros usuarios en los últimos meses.

- **Texto del *avatar*.** Para el campo del *avatar* hemos implementado una solución programada en C#, la cual nos proporciona el texto indicativo de la imagen del usuario, usando el nuevo servicio de *Google Imágenes*[2] y su capacidad de realizar consultas mediante el empleo de una imagen.

  El *avatar* es usado como una consulta del sistema y como representación hemos utilizado el texto de las imágenes más similares. Por ejemplo, en el caso del perfil de la Figura 3.22, hemos recuperado la ruta de la imagen

[1] `http://www.meneame.net/notame`
[2] `http://images.google.es`

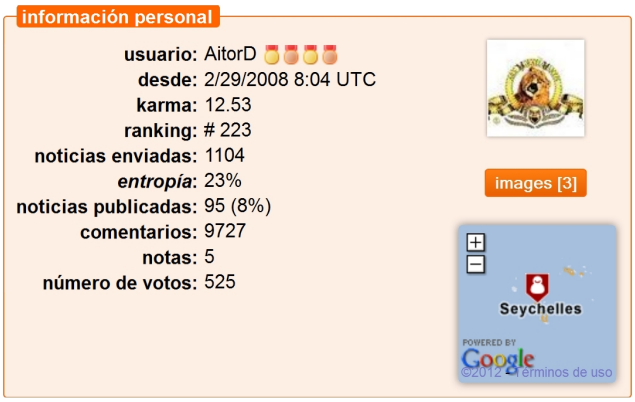



**Figura 3.22:** Perfil de un usuario de *Menéame*.

*avatar* del usuario y la hemos pegado como una consulta en el servicio de *Google Imágenes*, como muestra la Figura 3.23.


**Figura 3.23:** Uso del servicio de *Google Imágenes* con la ruta del *avatar* del usuario.

Después, el servicio recupera las imágenes más similares y la consulta más probable para esa imagen, como indica la Figura 3.24.

En el caso de que el servicio de *Google Imágenes*, no encuentre ninguna consulta posible para la imagen propuesta, nuestra solución deja en blanco el campo perteneciente al *avatar*. Además, existe un gran número de usuarios que no cambian el *avatar* por defecto de *Menéame* (el logotipo de la página web). Cuando esto sucede, con el fin de minimizar la sobrecarga de computación, nuestro sistema automáticamente rellena el campo *avatar* con el texto Meneame.

Gracias a este procedimiento, hemos podido usar la información contenida en el cuerpo del *avatar* empleando VSM (Baeza-Yates y Ribeiro-Neto, 1999), el cual representa documentos matemáticamente como vectores


**Figura 3.24:** Recuperación de imágenes similares a la consulta realizada.

en un espacio multidimensional. Este modelo puede ser representado utilizando palabras o *n-gramas* como término.

Estas características, como se menciona anteriormente, son obtenidas de la información pública del perfil del usuario y dotan al sistema de información general sobre el usuario en cuestión.

#### 3.3.2.2 Propiedades extraídas de los comentarios

Asimismo, hemos analizado los comentarios para contar el número de comentarios de cada categoría y poder moderar de mejor manera al usuario en cuestión, en base a sus características del perfil y a la clase de comentarios que expresa públicamente.

En la sección 3.2.3.1, hemos explicado como hemos llevado a cabo la clasificación de los comentarios en 3 categorías diferentes, y que retomamos a continuación:

- **Enfoque del comentario.** El comentario puede enfocarse a la noticia o a otros comentarios.
- **Tipo de comentario.** Se puede diferenciar entre aporte, irrelevante u opinión a la noticia o a otro comentario.
- **Grado del comentario.** Puede ser normal, polémico, muy polémico o gracioso.

Por su parte, el enfoque empleado utiliza diferentes características sintácticas, estadísticas y de opinión para construir una representación de los comentarios, explicadas en la sección 3.3.1 y que repasamos a continuación:

- **Estadísticas.**
  - Cuerpo del comentario.
  - Número de referencias al comentario.
  - Número de referencias desde el comentario.
  - Número de comentario en la noticia.
  - Similitud del comentario respecto al resumen de la noticia.
  - Número de coincidencias entre las palabras del comentario y las etiquetas de la noticia.
  - Número de enlaces en el cuerpo del comentario.
- **Sintácticas.**
  - Número de adjetivos en el cuerpo del comentario.
  - Número de números en el cuerpo del comentario.
  - Número de fechas en el cuerpo del comentario.
  - Número de adverbios en el cuerpo del comentario.
  - Número de nombres en el cuerpo del comentario.
  - Número de conjunciones en el cuerpo del comentario.
  - Número de pronombres en el cuerpo del comentario.
  - Número de signos de puntuación en el cuerpo del comentario.
  - Número de interjecciones en el cuerpo del comentario.
  - Número de determinantes en el cuerpo del comentario.
  - Número de abreviaturas en el cuerpo del comentario.
  - Número de preposiciones en el cuerpo del comentario.
  - Número de verbos en el cuerpo del comentario.
- **De opinión.**
  - Número de palabras positivas y negativas.
  - Número de votos.
  - *Karma*.

Una vez etiquetados y categorizados los comentarios, los hemos empleado como una característica más en el conjunto de usuarios, para poder representar a cada usuario con sus comentarios realizados dentro del periodo que comprende el conjunto de datos (del 5 de Abril al 12 de Abril de 2011). Por lo tanto, hemos generado 9 nuevas características que contabilizan el número de comentarios realizados por el usuario en estas categorías:

- **Número de comentarios referentes a la noticia.** Establece el número de comentarios de un usuario que comenta la noticia.
- **Número de comentarios referentes a otros comentarios.** Contabiliza el número de comentarios del usuario hacia otro comentario.
- **Número de comentarios aporte.** Cuenta el número de comentarios que pertenecen a la categoría *Aporte*.
- **Número de comentarios irrelevantes.** Calcula el número de comentarios de tipo irrelevante.
- **Número de comentarios de opinión.** Computa el número de comentarios del usuario que son opinión personal del mismo.
- **Número de comentarios normales.** Establece el número de comentarios que no son comentarios polémicos o graciosos.
- **Número de comentarios polémicos.** Contabiliza el número de comentarios que son polémicos.
- **Número de comentarios muy polémicos.** Calcula el número de comentarios que son muy polémicos respecto a la noticia o a otros comentarios.
- **Número de comentarios graciosos.** Computa el número de comentarios graciosos que realiza el usuario en sus publicaciones.

### 3.3.3 Discusión de los resultados

Después de adaptar tanto los 9.044 comentarios como los 3.359 usuarios, hemos construido 10 archivos ARFF (acrónimo del inglés *Attribute Relation File Format*) (Holmes *et al.*, 1994) con el vector resultante de las representaciones de los comentarios (6 archivos ARFF) y de los usuarios (4 archivos ARFF): (i) uno por cada clasificación y (ii) otro para cada tipo de VSM, con el fin de emplear

la herramienta de aprendizaje automático WEKA (acrónimo del inglés *Waikato Environment for Knowledge*) (Garner *et al.*, 1995).

En el caso del conjunto de comentarios disponemos de 3 clasificaciones:

- El tipo de comentario.
- El enfoque del comentario.
- El grado del comentario.

Mientras que en el conjunto de datos de usuarios hemos etiquetado en base a 2 categorías:

- El tipo de usuario.
- El grado del usuario.

Por su parte, en el caso de VSM hemos empleado 2 enfoques:

- La palabra.
- El *n-grama*.

Por lo tanto, aplicando cada enfoque de VSM a cada categoría de ambos conjuntos de datos, tenemos como resultado 6 archivos ARFF para el conjunto de comentarios y 4 para el conjunto de usuarios, lo cual ofrece un total de 10 archivos ARFF:

- El tipo de comentario con el VSM basado en palabras.
- El tipo de comentario con el VSM basado en *n-gramas*.
- El enfoque del comentario con el VSM basado en palabras.
- El enfoque del comentario con el VSM basado en *n-gramas*.
- El grado del comentario con el VSM basado en palabras.
- El grado del comentario con el VSM basado en *n-gramas*.
- El tipo de usuario con el VSM basado en palabras.
- El tipo de usuario con el VSM basado en *n-gramas*.

- El grado del usuario con el VSM basado en palabras.
- El grado del usuario con el VSM basado en *n-gramas*.

Una vez realizado el proceso de aplicación de VSM a ciertas características de nuestros conjuntos de datos y la creación de los archivos, la Tabla 3.6 muestra los resultados logrados para los diferentes conjuntos de datos.

**Tabla 3.6:** Número de características para cada conjunto de datos.

| **Conjunto de datos** | **# Características con palabras** | **# Características con *n-gramas*** |
|---|---|---|
| Comentarios | 13.247 | 34.126 |
| Usuarios | 2.120 | 3.600 |

Se puede observar el gran aumento de características debido a la aplicación tanto del enfoque de palabra basado en VSM como el enfoque de *n-grama* basado también en VSM, sobre el cuerpo del comentario y sobre el *avatar* del usuario. Principalmente, esto se debe a que hemos desglosado el texto de ambos atributos en las palabras más relevantes y en los *unigramas*, *bigramas* y *trigramas* más destacados, respecto a ambos enfoques.

Hemos empleado esta técnica basada en VSM y el esquema de ponderación TF-IDF, ya que en nuestro caso nos ofrece una serie de ventajas que favorecen el modelado. En el caso de TF-IDF, el procedimiento de TF clasifica los términos por mayor importancia, debido a su mayor frecuencia en un comentario, lo cual nos permite filtrar los términos con menor importancia en un comentario. Un problema de esta técnica, viene derivado del uso de comentarios extensos, cuya solución sería normalizarlos. Pero en nuestro caso, estos comentarios no son excesivamente extensos como para tener en cuenta este factor.

En el caso de IDF, también es necesario su aplicación, debido a que nos devuelve los términos más importantes de una colección, según más infrecuentes sean. Esto se debe a que este tipo de términos discriminan antes, con lo que definimos la *rareza* de un término. Por lo cual, gracias al empleo conjunto de ambos procedimientos, denominado TF-IDF, obtenemos una recuperación ordenada. Es decir, si un comentario es devuelto antes que otro, es debido a que dicho comentario es más relevante. Aunque por otra parte, se puede dar el caso del acoplamiento parcial, el cual indica que el comentario más relevante no tiene por qué contener todos los términos de la consulta. Pero no tiene por qué haber ningún comentario de este tipo, aunque hubiera comentarios relevantes.

Por último, el esquema TF-IDF es bastante intuitivo y a su vez, asume la independencia de términos (más conocido como *bag of words*).

Por otra parte, gracias al empleo de *n-gramas*, en nuestro caso *unigramas*, *bigramas* y *trigramas*, podríamos incrementar el contexto del estudio aumentando el tamaño de *n*, a fin de experimentar con nuevos indicadores a la espera de mejorar los resultados. Al *tokenizar* los comentarios en *n-gramas*, en vez de palabras, se consigue que el modelo sea independiente del dominio del idioma, ya que no integra ningún tipo de recurso concreto sobre ellos. Al contrario, el empleo de instancias a nivel de *n-grama* constituye en sí mismo un mecanismo de normalización de términos que permite, además, trabajar con una gran variedad de lenguas sin procesamiento alguno (Mcnamee y Mayfield, 2004; McNamee y Mayfield, 2004; Robertson y Willett, 1998).

## 3.4 **Sumario**

En este capítulo, hemos explicado el procesado inicial de datos realizado, así como el modelado o adaptación de dichos datos llevado a cabo, para poder clasificarlos correctamente en el siguiente paso de nuestro sistema de moderación automática y categorización.

En primer lugar, hemos desarrollado una introducción, con el fin de establecer los conceptos expuestos posteriormente en el capítulo. A continuación, hemos expuesto el sitio web social elegido sobre el cual se desarrolla esta investigación: *Meneame.net*, así como su funcionamiento, características y entrando en detalle, la estructura de las noticias y sus comentarios.

Más adelante, hemos procedido a explicar el procesado inicial de los datos empleado con el fin de normalizar todas las noticias y comentarios para poder categorizarlos correctamente. Para ello, primero hemos explicado el proceso automático de descarga, mediante el cual hemos obtenido los datos necesarios. Gracias a este proceso, hemos logrado el conjunto de comentarios que, posteriormente, es empleado para conseguir el siguiente conjunto de datos: el conjunto de usuarios y sus comentarios. Ya descritos y creados ambos conjuntos, hemos procedido a etiquetarlos en diversas categorías instauradas previamente. De este procesado inicial de los datos hemos obtenido una serie de resultados.

En último lugar, hemos desarrollado la adaptación de los datos pertenecientes a ambos conjuntos de datos, con el fin de optimizar estos conjuntos y eliminar datos inconexos o erróneos. Para ello, hemos comenzado por extraer las características de lo comentarios, las cuales hemos dividido en: (i) propiedades

estadísticas, (ii) propiedades sintácticas y (iii) propiedades de opinión; para más adelante, continuar con la extracción de características de los usuarios: (i) características extraídas del perfil y (ii) características extraídas de los comentarios del propio usuario. Al finalizar este tratamiento de los datos, hemos mostrado los resultados que nos ofrecen ambos conjuntos de datos.

*«Hay que buscar la verdad y no la razón de las cosas. Y la verdad se busca con humildad.»*

Miguel de Unamuno (1864–1936)

CAPÍTULO

# 4

# Categorización automática de comentarios y usuarios

El creciente interés por el procesamiento automático de opiniones, en diferentes documentos textuales, es una de las consecuencias del incremento exponencial del contenido generado por parte de los usuarios en lo denominado Web 2.0. Así como el interés de organizaciones, empresas o administraciones públicas, en analizar, filtrar o detectar de manera automática las opiniones publicadas por los diferentes tipos de usuarios en la red.

Debido a este creciente aumento, existe una disponibilidad de información digital, la cual ha permitido que la categorización automática sea algo accesible y por lo tanto, el desarrollo de diferentes técnicas para su implementación.

En este particular escenario, diferentes sitios web sociales como *Digg*[1], o su versión española *Menéame*[2], son ejemplos de este tipo de sitios que permiten a los usuarios enviar noticias para que otros usuarios puedan leerlas, votarlas o comentarlas. Normalmente, el criterio de publicación está gestionado en base a los votos, donde otros usuarios del sistema valoran cada noticia mediante sus votos y al final, las noticias más votadas por los usuarios aparecen en la portada (Lerman, 2007). Particularmente, *Menéame* ya tiene un método para valorar a

[1] `http://digg.com`
[2] `http://www.meneame.net`

sus usuarios y para moderar automáticamente los comentarios y noticias enlazadas. Este método está basado en los comentarios y votos de otros usuarios, con lo cual no es objetivo ya que podrían aparecer grupos de presión, y usuarios o grupos trol desvirtuando el sistema.

Para evitar estos problemas, en el contexto de la categorización de usuarios, se han realizado diferentes trabajos empleando las características de las pulsaciones del usuario en el teclado (Ypma y Heskes, 2003) o su actividad en los registros (Elahi *et al.*, 2010; Nasraoui *et al.*, 2008). Sin embargo, no existen trabajos específicos que se encarguen de la problemática de las noticias sociales y sus peculiaridades.

Para intentar paliar estas cuestiones, nuestras contribuciones más importantes son las siguientes:

- Un nuevo método basado en aprendizaje automático para categorizar usuarios y sus comentarios en sitios web sociales.
- Una demostración de que este método puede lograr altas tasas de precisión en 3 tareas de clasificación diferentes con la información sobre los comentarios extraída de *Menéame*.
- Una demostración de que este procedimiento puede alcanzar altos índices de acierto en 2 clasificaciones diferentes con los usuarios extraídos de *Menéame*.

El resto del capítulo está estructurado de la siguiente manera. La sección 4.1 revisa los diferentes clasificadores y métodos de aprendizaje supervisado utilizados en esta investigación. La sección 4.2 desarrolla la validación empírica para nuestra metodología, refiriéndose a la categorización de comentarios y usuarios en sitios web sociales, mediante el uso de los clasificadores de aprendizaje supervisado. La sección 4.3 hace referencia a los resultados y a las conclusiones obtenidas en este capítulo, en referencia a la clasificación de comentarios y usuarios en el sitio web social de noticias *Menéame*. Por último, la sección 4.4 resume los conceptos y las contribuciones expresadas a lo largo de este capítulo.

## 4.1 El aprendizaje automático

El investigador Mitchell (1997) proporcionó una definición formal sobre el significado de los términos aprendizaje automático:

> «*Un programa de ordenador se dice que aprende de la experiencia E con respecto a una clase de tareas T y una medida de rendimiento P, si su rendimiento en tareas de T, según la medida P, mejora con la experiencia E.*»

Como ejemplo explicativo proponemos el caso de aprender a reconocer la escritura manual por parte de una máquina, donde:

- **Tarea (T).** Implica reconocer y clasificar palabras manuscritas en una imagen.
- **Medida de rendimiento (P).** Hace referencia al porcentaje de palabras reconocidas correctamente.
- **Experiencia (E).** Se refiere a la base de datos de imágenes de palabras manuscritas, ya clasificadas.

El aprendizaje automático (ML, de las voces inglesas *Machine Learning*) es un campo derivado de la inteligencia artificial (AI, del inglés *Artificial Intelligence*), el cual se encarga de estudiar el problema del aprendizaje en las máquinas. Es decir, de cómo las máquinas pueden adquirir el conocimiento que las capacite para resolver determinados problemas (Mitchell, 1997). El ML está dirigido a automatizar el proceso de aprendizaje, de modo que el conocimiento pueda ser conseguido con una dependencia mínima de los hombres. Por lo tanto, el sistema puede aprender, bien adquiriendo nuevo conocimiento, bien modificando el conocimiento ya adquirido de manera más efectiva. El efecto de este aprendizaje es proporcionar a la máquina (o a la persona) un nuevo conocimiento, el cual le permita solucionar una mayor cantidad de problemas, lograr soluciones más precisas, obtener soluciones a más bajo costo o simplificar el conocimiento almacenado.

Existe una amplia variedad de aplicaciones en las cuales se han empleado técnicas de aprendizaje automático: diagnósticos médicos (Rahman *et al.*, 2007), clasificación de secuencias de ADN (García-Jiménez *et al.*, 2011), reconocimiento del habla y del lenguaje escrito (Srinivasan y Wang, 2006), robótica y juegos (deGroote, 2005), motores de búsqueda (Joachims y Radlinski, 2007), análisis del mercado de valores (Hung y Xu, 1999), predicción de series temporales (Peralta *et al.*, 2008), detección de defectos en piezas de fundición (Pastor-López *et al.*, 2012), etc.

A su vez, existen diversas clases de aprendizaje automático, las cuales detallamos a continuación:

- **Aprendizaje supervisado.** Este tipo de aprendizaje (Reed y Marks, 1998) consiste en un tipo de aprendizaje automático en donde al algoritmo que se usa para obtener el modelo relacional entre la entrada y las salidas, se le proporcionan una serie de ejemplos etiquetados que indican a qué clase pertenecen o el valor de salida, en el caso de problemas de regresión. Es decir, todos los ejemplos han sido clasificados a *priori*. De esta forma, en el proceso de aprendizaje, el algoritmo compara su salida actual con la etiqueta del ejemplo, para luego realizar los cambios que sean necesarios.

- **Aprendizaje no supervisado.** El aprendizaje no supervisado es un método de aprendizaje automático, en el que un modelo se ajusta a las observaciones dadas. La diferencia con el aprendizaje supervisado es que no existe un conocimiento previo. En el aprendizaje no supervisado no existe esa información que hace referencia a las salidas de los patrones que se van introduciendo, únicamente existen las entradas. Debe de ser el modelo, el cual vaya adecuando sus pesos dependiendo de la información interna que vaya recopilando de las entradas. Principalmente, esta clase de entrenamiento se emplea para clasificar y diferenciar rasgos significativos de un conjunto de datos no clasificado a *priori*, debido a que la red, internamente, intenta encontrar redundancias y rasgos significativos para agrupar los datos. Algunos algoritmos de agrupamiento o *clustering* como COBWEB (Fisher, 1987), EM (Dempster *et al.*, 1977), o $K$-Medias (Lloyd, 1982), son ejemplos de esta clase de aprendizaje.

En el desarrollo de esta investigación, hemos empleado el aprendizaje supervisado, ya que el conjunto de datos procedente de *Menéame*, tanto los comentarios como los usuarios, fueron etiquetados previamente.

### 4.1.1 Las redes bayesianas

Las redes bayesianas son modelos gráficos que se emplean para representar las relaciones entre los eventos o ideas, con el fin de inferir probabilidades e incertidumbres asociadas a dichos eventos o ideas. Estas redes están basadas en las teorías probabilísticas creadas en el siglo XVIII por el teólogo y matemático inglés *Thomas Bayes* y son útiles cuando se trata de predecir las relaciones probables. Las redes bayesianas se usan también en los algoritmos de búsqueda, dentro de una gran gama de aplicaciones que requieren recuperación de información, en el software de reconocimiento de imágenes y texto, etc.

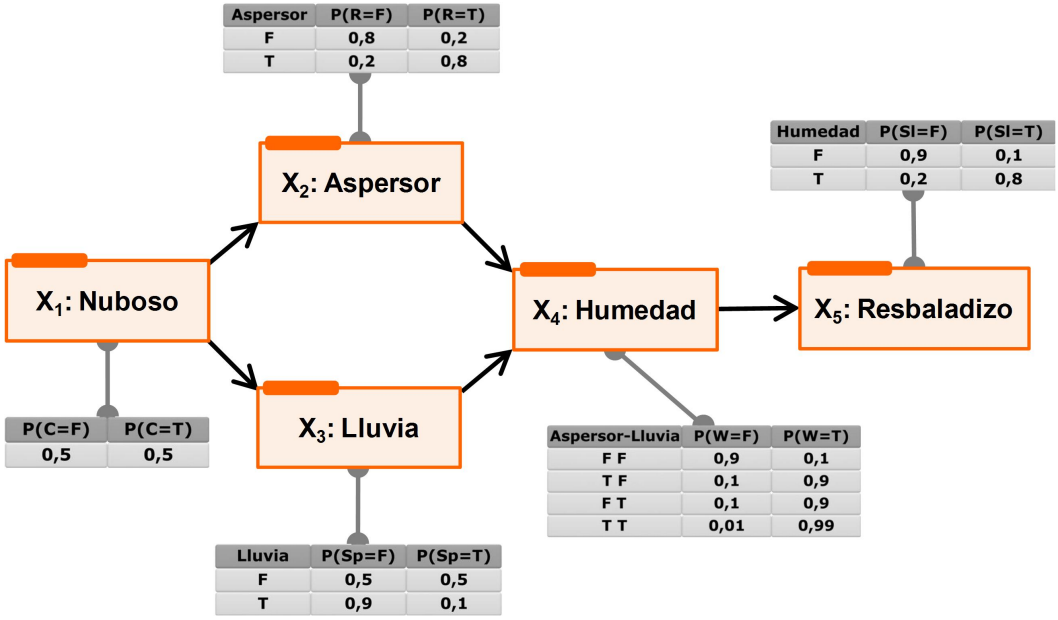



**Figura 4.1:** Ejemplo de clasificación mediante una red bayesiana.

Observando su formulación, se puede afirmar que dados dos eventos a los que denominaremos $A$ y $B$, la probabilidad condicionada de que ocurra $A$ habiéndose dado $B$, es decir $P(A|B)$, se puede obtener mediante la ecuación 4.1.

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \tag{4.1}$$

Gracias a este enfoque, se determinan las redes bayesianas (véase Figura 4.1), las cuales son modelos estadísticos basados en el teorema de Bayes (Pearl, 1982) y se definen como un modelo gráfico probabilístico para el aprendizaje y la predicción. Específicamente, las redes bayesianas se definen como un grafo acíclico dirigido[1] con una función de distribución probabilística asociada (Castillo *et al.*, 1997). Los nodos contenidos en el grafo dirigido ser refieren a las características de un problema concreto, mientras que las aristas representan las diferentes dependencias condicionales entre ellos. La función de probabilidad ilustra la robustez de las aristas en el grafo (Castillo *et al.*, 1997).

Por su parte, existen diferentes algoritmos de aprendizaje basados en las redes bayesianas y que hemos empleado en esta investigación:

[1]Un grafo dirigido o digrafo es un tipo de grafo en el cual el conjunto de las aristas tiene una dirección definida, a diferencia del grafo generalizado, en el cual la dirección puede estar especificada o no.

- **El algoritmo de aprendizaje K2.** El algoritmo K2 (Cooper y Herskovits, 1991a) se encarga de buscar la red estructural con mayor probabilidad dado un conjunto de datos.

- **El clasificador *Naïve Bayes*.** Un clasificador bayesiano ingenuo (Lewis, 1998) asume la inexistencia de cualquier tipo de dependencia entre los nodos o atributos de la propia red bayesiana. Por lo cual, solamente se tienen que aprender las probabilidades de los valores de los atributos, dada la clase.

- **El método *Tree Augmented Naïve* (TAN).** El método TAN (Friedman *et al.*, 1997) es un clasificador bayesiano simple aumentado con un árbol, el cual es una manera de mejorar la estructura del clasificador *Naïve Bayes* mediante la incorporación de ciertos nodos o atributos que tengan cierta dependencia.

### 4.1.2 Las máquinas de soporte vectorial

Las máquinas de soporte vectorial, también conocidas como SVM (acrónimo del inglés *Support Vector Machines*), dividen la representación del espacio *n*-dimensional de los datos en 2 regiones empleando un hiperplano, el cual intenta maximizar el margen entre las 2 regiones del espacio. Este margen es la distancia más lejana entre las instancias de entrenamiento de ambas clases y es calculada basándose en la distancia entre las instancias más cercanas al margen de ambas clases, que se denominan *vectores soporte* (Cortes y Vapnik, 1995) (véase la Figura 4.2).

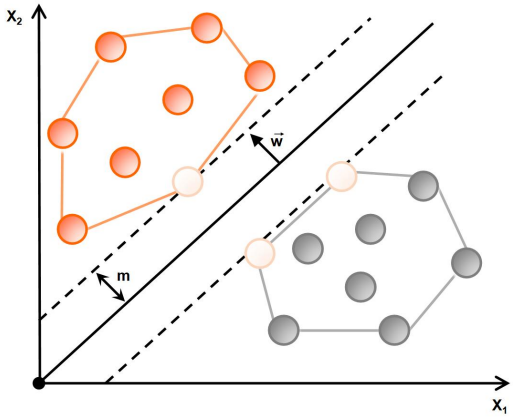



**Figura 4.2:** Ejemplo de clasificación mediante una máquina de soporte vectorial.

Formalmente, podemos definir un conjunto de datos de entrenamiento $\mathscr{D}$, de tal forma que:

$$\mathscr{D} = \{(x_i, c_i) x_i \in \mathbb{R}^p, c_i \in \{-1, 1\}\}_{i=1}^{n} \tag{4.2}$$

donde $c_i$ indica la clase del punto $x_i$, pudiendo ser su valor -1 o 1, y siendo a su vez, un vector de números reales en un espacio $p$-dimensional. El objetivo de este algoritmo es encontrar el hiperplano que maximice el margen $m$ y que divida los puntos $c_i = 1$ de los puntos $c_i =$ -1 en el espacio formado por el conjunto de datos $\mathscr{D}$. La representación del hiperplano se realiza mediante los puntos $x_i$, de tal forma que después de realizar el producto escalar entre $x$ y $w$, y restado el valor de $m$, el valor resultante sea 0, siendo $w$ un vector normal y ortogonal al hiperplano. Dicho de otro modo, los puntos $x_i$ deben cumplir la ecuación 4.3.

$$w \cdot x - m = 0 \tag{4.3}$$

De este modo se dispone de dos hiperplanos, correspondientes cada uno de ellos a cada clase de $c_i$, pudiéndose describir mediante la ecuación 4.4.

$$c_i(w \cdot x_i - m) \geq 1, \forall x_i 1 \leq i \leq n \tag{4.4}$$

En lugar de utilizar hiperplanos lineales, lo más común es emplear las funciones de núcleo que llevan a clasificaciones no lineales como polinomiales, radiales o sigmoidales (Amari y Wu, 1999). En particular, hemos utilizado las siguientes configuraciones del núcleo de las máquinas de soporte de vectorial, a la hora de realizar los experimentos pertenecientes a la investigación:

- **El núcleo polinomial.** Este tipo de núcleo representa la similitud de los vectores (instancias de entrenamiento) en un espacio característico sobre los polinomios de las variables originales (Amari y Wu, 1999).
- **El núcleo polinomial normalizado.** El núcleo normalizado es una medida de la distancia entre 2 puntos de la superficie de una esfera controlado por un parámetro, análogo a la distancia *Mahalanobis*[1]. Esta interpretación es particularmente interesante cuando se usa con la proyección equidistante, en la cual la longitud del arco entre 2 puntos de la esfera es la

[1]La distancia *Mahalanobis* es una forma de determinar la similitud entre 2 variables aleatorias multidimensionales. Se diferencia de la distancia euclídea en que considera la correlación entre las variables aleatorias.

misma que su separación en el plano medido por la distancia euclídea (Wan y Campbell, 2000).

- **El núcleo basado en funciones de base radial.** Este núcleo mapea muestras de forma no lineal en un espacio dimensional superior; de modo que, a diferencia del núcleo lineal, es capaz de gestionar la instancia cuando la relación entre las etiquetas de las clases y los atributos es no lineal (Hsu *et al.*, 2003). Hay algunas situaciones donde el núcleo RBF no es adecuado. En particular, cuando el número de características es muy alto.

- **El núcleo *Pearson VII*.** Basado en la función *Pearson VII*[1], se caracteriza por su robustez y gran potencia de mapeo (comparada con los núcleos anteriores), con lo que puede ser usada como un núcleo universal capaz de ser una alternativa genérica a las funciones comunes de núcleo (Üstün *et al.*, 2007).

### 4.1.3 Los $K$-vecinos más cercanos

El algoritmo $K$-vecinos más cercanos (KNN, del inglés *K-Nearest Neighbours*) (Fix y Hodges Jr, 1952) es un modelo de aprendizaje automático supervisado basado en instancias. Este método clasifica una instancia desconocida basándose en la clase de las instancias más cercanas a dicha instancia, en el espacio de entrenamiento. Para este fin, la instancia desconocida se coloca en este espacio y el algoritmo mide la distancia entre las instancias de entrenamiento y esta instancia desconocida. A pesar de los diferentes métodos existentes a la hora de elegir la clase de la instancia, la técnica más común es simplemente clasificarla como la clase más común entre los $K$-vecinos más cercanos.

A continuación desarrollamos el concepto de matemático del algoritmo $K$-vecinos más cercanos. En este clasificador basado en instancias, la fase de entrenamiento se inicia empleando un conjunto de instancias $\mathscr{S} = \{s_1, s_2, ..., s_{n-1}, s_n\}$, donde $n$ representa el número de características de cada una de las instancias. Este conjunto de datos se representa en un espacio de entrenamiento como el mostrado en la Figura 4.3.

Más adelante, y con la llegada de una nueva instancia a categorizar, comienza la fase de clasificación. Llegados a este punto, se calcula la distancia de la nueva instancia respecto a las instancias utilizadas en la fase de entrenamiento.

[1]La función *Pearson VII* era una función popular en los años 1980 y 1990 para describir formas picudas a partir de patrones de difracción de rayos X convencionales.

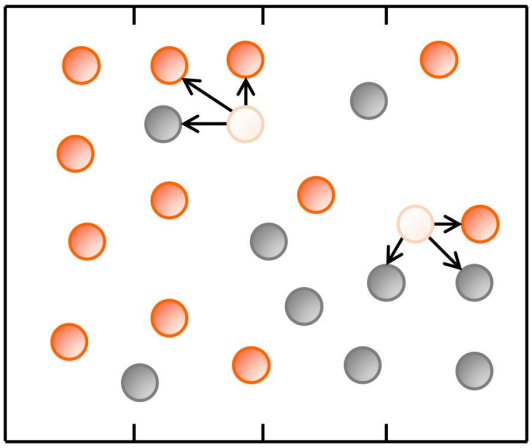


**Figura 4.3:** Ejemplo de clasificación mediante $K$-vecinos más cercanos.

Dicha distancia puede ser calculada mediante distintos tipos de métricas, como por ejemplo: (i) la distancia euclidiana[1], (ii) la distancia del coseno[2] o (iii) la distancia *Manhattan*[3], entre otras. En esta tesis hemos empleado la distancia euclidiana, cuya definición matemática es la mostrada en la ecuación 4.5.

$$\sqrt{\sum_{i=0}^{n}(X_i - Y_i)^2} \tag{4.5}$$

Una vez calculadas las distancias, en base a la métrica elegida previamente, podemos emplear distintos métodos para determinar la clase de la instancia desconocida. La técnica más empleada es elegir la clase más común entre las $K$ instancias más cercanas pertenecientes al espacio de entrenamiento y la instancia a clasificar.

### 4.1.4 Los árboles de decisión

Los clasificadores de árboles de decisión (DT, de las voces inglesas *Decision Trees*) son un tipo de algoritmos pertenecientes al aprendizaje automático supervisado,

[1]La distancia euclidiana o euclídea es la distancia *ordinaria* entre 2 puntos de un espacio euclidiano; es decir, la distancia que se mediría con una regla.

[2]La distancia del coseno es una medida de similitud entre 2 vectores de un espacio vectorial que mide el coseno del ángulo formado entre ellos.

[3]La distancia *Manhattan* es la distancia entre 2 puntos medida sobre los ejes a ángulos rectos; es decir, la distancia que se recorrería para llegar de un punto a otro si se siguiera una trayectoria cuadriculada.

los cuales son representados gráficamente en forma de árboles (véase la Figura 4.4). En este tipo de algoritmos, los nodos internos representan las condiciones de prueba con las variables de un problema concreto, mientras que los nodos finales u hojas representan la decisión final del algoritmo (Quinlan, 1993).

Para conocer la estructura de árbol de estos modelos de un conjunto de datos etiquetado, existen diferentes métodos de entrenamiento a utilizar. En particular, hemos empleado los siguientes:

- **El algoritmo C4.5.** En la herramienta WEKA (acrónimo del inglés *Waikato Environment for Knowledge*), la implementación de este algoritmo se denomina J48 (Quinlan, 1993). Este clasificador es una extensión del algoritmo ID3 (Quinlan, 1986). Su funcionamiento se basa en un concepto denominado *entropía de la información*, la cual calcula la incertidumbre asociada a una variable aleatoria. En ocasiones, este término va asociado a la *entropía de Shannon* (Shannon, 1951), que mide el valor esperado de información dentro de un mensaje.

- **Los bosques aleatorios.** Otro método empleado son los bosques aleatorios (RF, del inglés *Random Forest*) (Breiman, 2001). Este algoritmo genera varios árboles de decisión de forma aleatoria, combinándolos posteriormente como si fuesen un único clasificador. Concretamente, se genera un vector de entrada, con cada uno de estos árboles, formado por un subconjunto de datos elegidos mediante una discriminación estocástica (Kleinberg, 1996).

A continuación, desarrollamos el concepto matemático del algoritmo de árboles de decisión. Formalmente, se puede definir un árbol de decisión como un grafo $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, donde $\mathcal{V}$ es un conjunto de nodos no vacío y $\mathcal{E}$ es un conjunto de aristas. Por lo cual, se define el camino del árbol como la secuencia de aristas que recorre, pudiendo ser representado de la forma $(v_1, v_2), (v_2, v_3), ..., (v_{n-1}, v_n)$. Por lo tanto, si tenemos en cuenta que $(v, w)$ es una arista del árbol, se puede definir $v$ como el nodo padre de $w$, y a su vez $w$ como el nodo hijo de $v$. El único nodo en el árbol sin padres se denomina nodo raíz. Cualquier otro nodo en el árbol es un nodo interno. Para construir la representación gráfica de un árbol se contesta a un conjunto de preguntas binarias (sí o no).

Más adelante, cada uno de los árboles generados realiza una clasificación. Para determinar el resultado final, se emplea un sistema de votos en el que participan todas las clasificaciones de los árboles generados.

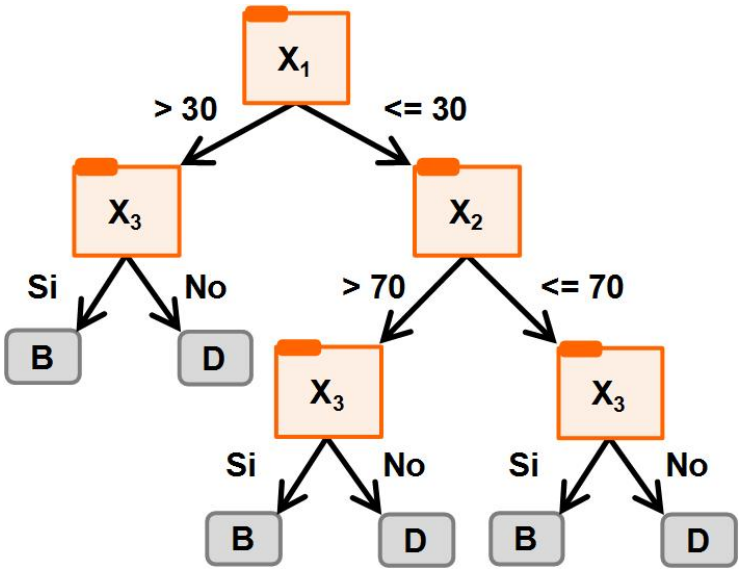



**Figura 4.4:** Ejemplo de clasificación mediante un árbol de decisión.

El principal inconveniente que presenta este clasificador es la sensibilidad al ruido que puedan presentar los nuevos datos (Svetnik *et al.*, 2004). Es decir, se adapta correctamente al conjunto de datos con los que realizamos el entrenamiento, pero pierde precisión en caso de que los futuros nuevos datos contengan ruido.

## 4.2 Validación empírica

En esta sección, se describe la validación que hemos llevado a cabo, con el fin de analizar la idoneidad de nuestra experimentación tanto en el caso de la categorización de comentarios, como en la categorización de usuarios.

### 4.2.1 Metodología

Una vez pre-procesados y modelados los conjuntos de datos de comentarios y usuarios en archivos tipo ARFF, hemos evaluado la precisión de nuestro método para categorizar los usuarios y sus comentarios al respecto. De este modo, por medio del conjunto de datos, hemos realizado la siguiente metodología:

- **Ganancia de Información.** Para cada uno de los tipos de clasificación, hemos extraído las características principales empleando la ganancia de

información (IG, de los términos ingleses *Information Gain*) (Kent, 1983), un algoritmo que evalúa la relevancia de un atributo midiendo la ganancia de información respecto a la clase:

$$IG(j) = \sum_{v_j \in R} \sum_{C_i} P(v_j, C_i) \cdot \frac{P(v_j, C_i)}{P(v_j) \cdot P(C_i)} \tag{4.6}$$

donde los términos que forman la ecuación, se explican a continuación:

- El término $C_i$ es la clase $i$-ésima.
- El término $v_j$ es el valor del atributo $j$.
- El término $P(v_j, C_i)$ es la probabilidad de que el atributo $j$ tenga el valor $v_j$ en la clase $C_i$.
- El término $P(v_j)$ es la probabilidad de que el atributo $j$ tenga el valor $v_j$ en el conjunto de datos de entrenamiento.
- El término $P(C_i)$ es la probabilidad de que el conjunto de datos de entrenamiento pertenezca a la clase $C_i$.

Empleando esta medida, hemos eliminado cualquier característica en el conjunto de entrenamiento que tenga un valor de IG $= 0$. Observamos los resultados obtenidos, después de aplicar este filtro, en la Tabla 4.1 y en la Tabla 4.2.

**Tabla 4.1:** Número de características para cada categorización en el conjunto de comentarios.

| **Categorización** | **# Características con palabras** | **# Características con *n-gramas*** |
|---|---|---|
| Enfoque del comentario | 648 | 1.209 |
| Tipo de comentario | 1.413 | 3.616 |
| Grado del comentario | 276 | 920 |
| Total | 2.337 | 5.745 |

Como se puede observar en la Tabla 4.3 y en la Tabla 4.4, la aplicación del filtro IG permite eliminar los términos innecesarios para la categorización, logrando una importante reducción de ruido, un aumento de la eficiencia e incluso un ligero crecimiento de la efectividad.

**Tabla 4.2:** Número de características para cada categorización en el conjunto de usuarios.

| **Categorización** | **# Características con palabras** | **# Características con *n-gramas*** |
|---|---|---|
| Grado del usuario | 21 | 21 |
| Tipo de usuario | 21 | 22 |
| Total | 42 | 43 |

**Tabla 4.3:** Reducción de características en el conjunto de comentarios mediante *Information Gain*.

| **Conjunto de datos** | **# Características con palabras** | **# Características con *n-gramas*** |
|---|---|---|
| Comentarios sin IG | 13.247 | 34.126 |
| Comentarios con IG | 2.337 | 5.745 |
| Reducción (%) | 82,35 % | 83,15 % |

**Tabla 4.4:** Reducción de características en el conjunto de usuarios mediante *Information Gain*.

| **Conjunto de datos** | **# Características con palabras** | **# Características con *n-gramas*** |
|---|---|---|
| Usuarios sin IG | 2.120 | 3.600 |
| Usuarios con IG | 42 | 43 |
| Reducción (%) | 98,01 % | 98,80 % |

- **SMOTE.** Los conjuntos de datos no están balanceados en las diferentes clases, debido a la heterogeneidad de los datos extraídos. Para hacer frente a los datos desbalanceados, hemos aplicado la técnica SMOTE (acrónimo de las voces inglesas *Synthetic Minority Over-sampling TEchnique*) (Chawla *et al.*, 2002), la cual es una combinación de sobre-muestreo de las clases menos pobladas y sub-muestreo de las más pobladas. El sobre-muestreo se realiza creando ejemplos sintéticos de la clase minoritaria de cada conjunto de datos de entrenamiento. La clase minoritaria es sobre-muestreada tomando cada muestra de la clase minoritaria e introduciendo ejemplos sintéticos a lo largo de los segmentos lineales uniendo con todos los $k$ ve-

cinos más cercanos de la clase minoritaria. Dependiendo de la cantidad de sobre-muestreo requerida, los vecinos pertenecientes a los *k* vecinos más cercanos son elegidos aleatoriamente. De esta manera, las instancias seguirán siendo únicas y las clases pasarán a estar equilibradas. También hemos ejecutado la evaluación de los algoritmos de aprendizaje automático sin aplicar SMOTE, con el fin de comparar los resultados y extraer conclusiones.

- **Validación cruzada.** La validación cruzada (del inglés *cross validation*) es una práctica estadística, la cual permite dividir un conjunto de datos en subconjuntos de tal manera que el análisis se realiza inicialmente en uno de ellos, mientras que los subconjuntos restantes son retenidos para su posterior empleo en la confirmación y validación de dicho análisis inicial. Explicado de otra manera, se basa en la división del conjunto de datos en *k* subconjuntos distintos, utilizándose una parte de cada uno de los subconjuntos para entrenar el modelo y la otra para testearlo.

  En nuestros experimentos, hemos asignado el valor de 10 a la variable *k*, con lo que hemos dividido el conjunto de datos total en 10 subconjuntos diferentes. En cada uno de los subconjuntos hemos empleado el 90 % para entrenar el modelo y el 10 % restante para probarlo. Este método se aplica generalmente en la evaluación del aprendizaje automático (Bishop, 1995).

- **Aprendizaje del modelo.** Para cada subconjunto de datos, hemos efectuado la etapa de aprendizaje de cada algoritmo utilizando diferentes parámetros o distintos tipos de algoritmos de aprendizaje, dependiendo del modelo específico. Los algoritmos usan los parámetros por defecto en la herramienta de aprendizaje automático WEKA. En particular, hemos usado los siguiente modelos:

  - *Redes bayesianas (BN, del inglés Bayesian Networks)*. Hemos utilizado diferentes algoritmos de aprendizaje estructural: K2 (Cooper y Herskovits, 1991b) y *Tree Augmented Naïve* (TAN) (Geiger *et al.*, 1997). Por otra parte, también hemos realizado experimentos con un clasificador *Naïve Bayes* (Bishop, 1995).
  - *Máquinas de soporte vectorial (SVM, del inglés Support Vector Machines)*. Hemos elaborado experimentos con un núcleo polinomial (del inglés *polynomial kernel*) (Amari y Wu, 1999), un núcleo polinomial normalizado (del inglés *normalized polynomial kernel*) (Maji *et al.*,

2008), un núcleo *Pearson VII* (Üstün *et al.*, 2007) y un núcleo basado en funciones de base radial (RBF, del inglés *radial basis function*) (Cho *et al.*, 2008).

- *K-vecinos más cercanos (KNN, del inglés K-nearest neighbour)*. Hemos realizado experimentos con diferentes valores de $K$, $K = 1$, $K = 2$, $K = 3$, $K = 4$ y $K = 5$.
- *Árboles de decisión (DT, del inglés Decision Trees)*. Hemos efectuado experimentos con J48 (la implementación de WEKA (Garner *et al.*, 1995) del algoritmo *C4.5* (Quinlan, 1993)) y *Random Forest*, un conjunto de árboles de decisión construidos al azar. Particularmente, hemos probado *Random Forest* con un número variable de árboles aleatorios $N$, $N = 10$, $N = 50$ y $N = 100$.

- **Evaluación del modelo.** Para probar el método, hemos empleado distintas métricas como: (i) la precisión del modelo (porcentaje de instancias clasificadas correctamente) y (ii) el área bajo la curva ROC (AUC, de las voces inglesas *Area Under the ROC Curve*). Para poder explicar estas medidas, es necesario introducir previamente los conceptos de TPR y FPR:

  - *Tasa de verdaderos positivos (TPR, de las voces inglesas True Positive Rate)*. Mide hasta qué punto un clasificador es capaz de detectar o clasificar los casos positivos correctamente, de entre todos los casos positivos disponibles. También se define como la sensibilidad.
  - *Tasa de falsos positivos (FPR, del inglés False Positive Rate)*. Define cuántos resultados positivos son incorrectos de entre todos los casos negativos disponibles. También conocido como especificidad.
  - *Precisión.* Es el número total de aciertos de los clasificadores dividido por el número total de casos en el conjunto de datos.
  - *Área bajo la curva ROC.* Es la representación gráfica de la sensibilidad frente a (1-especificidad). Similarmente, es la representación gráfica de la tasa de verdaderos positivos frente a la tasa de falsos positivos (Singh *et al.*, 2009). Este índice se puede interpretar como la probabilidad de que un clasificador ordenará una instancia positiva, elegida al azar, más alta que una instancia negativa.

## 4.3 Discusión de los resultados

Esta sección expone los resultados logrados mediante la experimentación realizada. Para ello, la metodología se ha divido en 2 fases. En la primera fase, hemos etiquetado y categorizado los comentarios descargados de *Menéame*; mientras que en la segunda fase, hemos procedido a etiquetar y clasificar a los usuarios en base al conjunto de comentarios correspondiente para cada usuario.

### 4.3.1 Conjunto de comentarios

**Tabla 4.5:** Resultados obtenidos en base a la precisión (%) y AUC, con y sin SMOTE, para el enfoque VSM utilizando palabras, respecto a la clasificación Enfoque del comentario.

| | Sin SMOTE | | Con SMOTE | |
|---|---|---|---|---|
| **Clasificador** | **Prec.(%)** | **AUC** | **Prec.(%)** | **AUC** |
| KNN K = 1 | 84,78 ± 1,00 | 0,83 ± 0,01 | 86,09 ± 0,95 | 0,85 ± 0,01 |
| KNN K = 2 | 78,17 ± 1,04 | 0,87 ± 0,01 | 81,18 ± 0,64 | 0,89 ± 0,01 |
| KNN K = 3 | 81,50 ± 1,06 | 0,89 ± 0,01 | 84,52 ± 0,97 | 0,91 ± 0,01 |
| KNN K = 4 | 78,13 ± 1,09 | 0,91 ± 0,01 | 81,31 ± 1,08 | 0,92 ± 0,01 |
| KNN K = 5 | 80,35 ± 1,15 | 0,92 ± 0,01 | 83,68 ± 1,21 | 0,93 ± 0,01 |
| BN: Bayes K2 | 92,48 ± 0,90 | 0,97 ± 0,01 | 93,00 ± 0,88 | 0,96 ± 0,01 |
| BN: Bayes TAN | 97,62 ± 0,51 | 0,99 ± 0,00 | 97,16 ± 0,65 | 0,99 ± 0,00 |
| Naïve Bayes | 84,11 ± 1,26 | 0,83 ± 0,01 | 84,55 ± 1,05 | 0,95 ± 0,01 |
| SVM: Polinomial | 96,72 ± 0,54 | 0,96 ± 0,01 | 96,97 ± 0,66 | 0,97 ± 0,01 |
| SVM: Polinomial normal. | 97,24 ± 0,52 | 0,97 ± 0,01 | 97,31 ± 0,73 | 0,97 ± 0,01 |
| SVM: PVK | 96,57 ± 0,57 | 0,96 ± 0,01 | 96,69 ± 0,96 | 0,97 ± 0,01 |
| SVM: RBF | 95,69 ± 0,57 | 0,95 ± 0,01 | 95,70 ± 0,79 | 0,95 ± 0,01 |
| DT: J48 | 95,79 ± 0,66 | 0,97 ± 0,01 | 95,62 ± 0,85 | 0,97 ± 0,01 |
| DT: Random Forest N=10 | 96,70 ± 0,65 | 0,99 ± 0,00 | 96,31 ± 0,63 | 0,99 ± 0,00 |
| DT: Random Forest N=50 | 97,39 ± 0,56 | 0,99 ± 0,00 | 97,38 ± 0,81 | 0,99 ± 0,00 |
| DT: Random Forest N=100 | 97,52 ± 0,54 | 0,99 ± 0,00 | 97,53 ± 0,72 | 0,99 ± 0,00 |

Las Tablas 4.5 y 4.6 muestran los resultados de la categorización respecto al enfoque del comentario. En general, los resultados son mejores cuando los términos empleados son palabras, que cuando se utilizan los *n-gramas*. Esta categorización es la más simple y los resultados muestran que el clasificador de redes bayesianas con el algoritmo de aprendizaje estructural TAN logra una precisión de 97,62% junto con un valor de AUC de 0,99. En este caso, se observa como las redes bayesianas, las máquinas de soporte vectorial y los árboles de decisión ofrecen resultados similares tanto en términos de porcentaje de precisión como de AUC, y a su vez mediante el empleo o no de la técnica SMOTE.

**Tabla 4.6:** Resultados logrados en cuanto a precisión (%) y AUC, con y sin SMOTE, para VSM utilizando *n-gramas*, respecto al Enfoque del comentario.

| | Sin SMOTE | | Con SMOTE | |
|---|---|---|---|---|
| **Clasificador** | **Prec.(%)** | **AUC** | **Prec.(%)** | **AUC** |
| KNN K = 1 | 78,98 ± 1,16 | 0,76 ± 0,01 | 80,43 ± 0,99 | 0,78 ± 0,01 |
| KNN K = 2 | 71,74 ± 1,16 | 0,81 ± 0,01 | 75,62 ± 1,62 | 0,84 ± 0,01 |
| KNN K = 3 | 75,41 ± 1,39 | 0,84 ± 0,01 | 79,59 ± 1,34 | 0,87 ± 0,01 |
| KNN K = 4 | 71,04 ± 1,34 | 0,86 ± 0,01 | 75,76 ± 1,16 | 0,88 ± 0,01 |
| KNN K = 5 | 73,41 ± 1,75 | 0,87 ± 0,01 | 78,91 ± 1,50 | 0,89 ± 0,01 |
| BN: Bayes K2 | 91,35 ± 1,28 | 0,95 ± 0,01 | 91,07 ± 0,77 | 0,94 ± 0,01 |
| BN: Bayes TAN | 96,78 ± 0,71 | 0,99 ± 0,00 | 95,75 ± 0,65 | 0,98 ± 0,01 |
| Naïve Bayes | 71,63 ± 0,67 | 0,69 ± 0,01 | 84,01 ± 1,16 | 0,82 ± 0,01 |
| SVM: Polinomial | 96,76 ± 0,62 | 0,96 ± 0,01 | 96,88 ± 0,59 | 0,96 ± 0,01 |
| SVM: Polinomial normal. | 96,31 ± 0,69 | 0,96 ± 0,01 | 96,34 ± 0,80 | 0,96 ± 0,01 |
| SVM: PVK | 95,31 ± 0,64 | 0,95 ± 0,01 | 95,48 ± 0,57 | 0,95 ± 0,01 |
| SVM: RBF | 95,75 ± 0,74 | 0,95 ± 0,01 | 95,80 ± 0,71 | 0,95 ± 0,01 |
| DT: J48 | 95,63 ± 0,95 | 0,96 ± 0,01 | 95,69 ± 0,92 | 0,97 ± 0,01 |
| DT: Random Forest N=10 | 95,59 ± 0,67 | 0,99 ± 0,00 | 94,95 ± 0,77 | 0,98 ± 0,01 |
| DT: Random Forest N=50 | 96,98 ± 0,82 | 0,99 ± 0,00 | 96,79 ± 0,79 | 0,99 ± 0,00 |
| DT: Random Forest N=100 | 97,13 ± 0,92 | 0,99 ± 0,00 | 96,99 ± 0,85 | 0,99 ± 0,00 |

**Tabla 4.7:** Resultados obtenidos en base a la precisión (%) y AUC, con y sin SMOTE, para el enfoque VSM utilizando palabras, respecto a la clasificación Tipo del comentario.

| | Sin SMOTE | | Con SMOTE | |
|---|---|---|---|---|
| **Clasificador** | **Prec.(%)** | **AUC** | **Prec.(%)** | **AUC** |
| KNN K = 1 | 65,48 ± 1,22 | 0,65 ± 0,01 | 60,74 ± 1,05 | 0,65 ± 0,01 |
| KNN K = 2 | 71,32 ± 0,98 | 0,70 ± 0,01 | 65,39 ± 1,21 | 0,68 ± 0,01 |
| KNN K = 3 | 65,91 ± 1,32 | 0,72 ± 0,02 | 55,40 ± 1,80 | 0,69 ± 0,02 |
| KNN K = 4 | 71,37 ± 1,35 | 0,73 ± 0,01 | 59,86 ± 1,52 | 0,70 ± 0,02 |
| KNN K = 5 | 66,45 ± 1,33 | 0,74 ± 0,01 | 51,55 ± 1,53 | 0,71 ± 0,02 |
| BN: Bayes K2 | 68,50 ± 1,19 | 0,82 ± 0,02 | 77,08 ± 1,58 | 0,83 ± 0,02 |
| BN: Bayes TAN | 74,56 ± 1,65 | 0,82 ± 0,02 | 78,22 ± 1,37 | 0,84 ± 0,02 |
| Naïve Bayes | 33,50 ± 8,56 | 0,78 ± 0,02 | 43,62 ± 1,08 | 0,75 ± 0,02 |
| SVM: Polinomial | 79,15 ± 1,48 | 0,77 ± 0,02 | 75,60 ± 1,47 | 0,79 ± 0,01 |
| SVM: Polinomial normal. | 79,58 ± 1,69 | 0,73 ± 0,03 | 77,21 ± 1,27 | 0,77 ± 0,02 |
| SVM: PVK | 79,47 ± 1,18 | 0,75 ± 0,02 | 77,24 ± 0,96 | 0,79 ± 0,01 |
| SVM: RBF | 70,82 ± 0,27 | 0,51 ± 0,00 | 66,97 ± 1,04 | 0,76 ± 0,01 |
| DT: J48 | 74,49 ± 1,90 | 0,72 ± 0,03 | 73,10 ± 1,84 | 0,74 ± 0,02 |
| DT: Random Forest N=10 | 76,92 ± 1,31 | 0,81 ± 0,01 | 76,51 ± 1,67 | 0,81 ± 0,02 |
| DT: Random Forest N=50 | 78,51 ± 1,27 | 0,83 ± 0,02 | 78,26 ± 1,47 | 0,84 ± 0,02 |
| DT: Random Forest N=100 | 78,87 ± 1,60 | 0,84 ± 0,02 | 78,58 ± 1,59 | 0,84 ± 0,02 |

**Tabla 4.8:** Resultados logrados en cuanto a precisión (%) y AUC, con y sin SMOTE, para VSM utilizando *n-gramas*, respecto al Tipo del comentario.

| | **Sin SMOTE** | | **Con SMOTE** | |
|---|---|---|---|---|
| **Clasificador** | **Prec.(%)** | **AUC** | **Prec.(%)** | **AUC** |
| KNN K = 1 | 52,27 ± 1,64 | 0,59 ± 0,02 | 48,32 ± 1,37 | 0,59 ± 0,01 |
| KNN K = 2 | 58,16 ± 1,68 | 0,61 ± 0,02 | 52,13 ± 1,02 | 0,60 ± 0,01 |
| KNN K = 3 | 45,28 ± 1,06 | 0,61 ± 0,01 | 38,61 ± 1,07 | 0,60 ± 0,01 |
| KNN K = 4 | 49,94 ± 1,24 | 0,61 ± 0,01 | 40,15 ± 1,03 | 0,61 ± 0,01 |
| KNN K = 5 | 41,90 ± 1,21 | 0,61 ± 0,01 | 35,01 ± 0,90 | 0,61 ± 0,01 |
| BN: Bayes K2 | 68,57 ± 1,61 | 0,83 ± 0,02 | 76,52 ± 1,93 | 0,85 ± 0,02 |
| BN: Bayes TAN | 73,05 ± 1,58 | 0,83 ± 0,02 | 78,84 ± 1,62 | 0,86 ± 0,01 |
| Naïve Bayes | 37,26 ± 7,76 | 0,68 ± 0,01 | 52,11 ± 1,42 | 0,67 ± 0,01 |
| SVM: Polinomial | 81,18 ± 1,18 | 0,80 ± 0,01 | 78,54 ± 1,10 | 0,81 ± 0,01 |
| SVM: Polinomial normal. | 80,46 ± 1,39 | 0,73 ± 0,02 | 80,13 ± 1,23 | 0,77 ± 0,01 |
| SVM: PVK | 81,93 ± 0,79 | 0,77 ± 0,01 | 81,15 ± 0,81 | 0,79 ± 0,01 |
| SVM: RBF | 70,92 ± 0,31 | 0,51 ± 0,00 | 76,86 ± 1,23 | 0,81 ± 0,01 |
| DT: J48 | 74,72 ± 2,42 | 0,74 ± 0,03 | 72,71 ± 2,02 | 0,74 ± 0,02 |
| DT: Random Forest N=10 | 78,13 ± 1,66 | 0,82 ± 0,02 | 77,75 ± 1,41 | 0,82 ± 0,02 |
| DT: Random Forest N=50 | 79,07 ± 1,58 | 0,85 ± 0,02 | 79,22 ± 1,60 | 0,84 ± 0,02 |
| DT: Random Forest N=100 | 79,51 ± 1,65 | 0,85 ± 0,02 | 79,85 ± 1,66 | 0,85 ± 0,02 |

Las Tablas 4.7 y 4.8 muestran los resultados sobre la categorización referente al tipo de información del comentario. En general, dichos resultados son también mejores cuando se utiliza las palabras como términos en el enfoque VSM, que cuando se usan los *n-gramas*. Sin embargo, el mejor resultado es obtenido por *Random Forest* con 100 árboles, cuando el contenido del cuerpo del comentario es *tokenizado* con *n-gramas*: 79,85 % en precisión y 0,85 de AUC. Destacamos que, aunque las máquinas de soporte vectorial, excepto el núcleo RBF, obtienen precisiones más elevadas que el clasificador *Random Forest* con *N* = 100, su valor de AUC es bastante menor que dichos clasificadores.

Las Tablas 4.9 y 4.10 presentan los resultados de la categorización de comentarios en referencia a su grado de controversia. En este caso, los resultados obtenidos reflejan una mejoría en el empleo de *n-gramas*, respecto a las palabras, en el enfoque VSM. Los mejores resultados son los logrados por el clasificador *Random Forest* con 100 árboles: 69,83 % de precisión y un valor de AUC de 0,70. Resaltamos que, al igual que en la clasificación en cuanto al tipo de comentario, los clasificadores SVM, empleando tanto palabras como *n-gramas* como términos, ofrecen mejores resultados en términos de precisión pero bastante inferiores en términos de AUC. Esta categorización es la más compleja de clasificar, ya que es altamente subjetiva.

**Tabla 4.9:** Resultados obtenidos en base a la precisión (%) y AUC, con y sin SMOTE, para el enfoque VSM utilizando palabras, respecto a la clasificación Grado del comentario.

| | Sin SMOTE | | Con SMOTE | |
|---|---|---|---|---|
| **Clasificador** | **Prec.(%)** | **AUC** | **Prec.(%)** | **AUC** |
| KNN K = 1 | 60,07 ± 1,83 | 0,59 ± 0,02 | 47,82 ± 1,94 | 0,59 ± 0,02 |
| KNN K = 2 | 68,04 ± 0,66 | 0,62 ± 0,02 | 55,20 ± 1,84 | 0,62 ± 0,02 |
| KNN K = 3 | 68,23 ± 0,67 | 0,63 ± 0,02 | 49,18 ± 1,63 | 0,63 ± 0,02 |
| KNN K = 4 | 69,73 ± 0,56 | 0,64 ± 0,02 | 49,69 ± 2,33 | 0,64 ± 0,03 |
| KNN K = 5 | 70,43 ± 0,45 | 0,64 ± 0,01 | 47,11 ± 1,76 | 0,64 ± 0,02 |
| BN: Bayes K2 | 50,13 ± 1,65 | 0,61 ± 0,02 | 69,99 ± 0,45 | 0,63 ± 0,03 |
| BN: Bayes TAN | 70,08 ± 0,56 | 0,64 ± 0,01 | 70,31 ± 0,40 | 0,66 ± 0,02 |
| Naïve Bayes | 25,94 ± 1,14 | 0,63 ± 0,02 | 33,55 ± 1,00 | 0,62 ± 0,02 |
| SVM: Polinomial | 72,62 ± 0,36 | 0,56 ± 0,01 | 45,40 ± 1,19 | 0,67 ± 0,01 |
| SVM: Polinomial normal. | 73,29 ± 0,29 | 0,59 ± 0,01 | 49,17 ± 1,34 | 0,64 ± 0,03 |
| SVM: PVK | 72,70 ± 0,47 | 0,57 ± 0,01 | 52,73 ± 1,68 | 0,64 ± 0,02 |
| SVM: RBF | 69,96 ± 0,05 | 0,50 ± 0,00 | 34,20 ± 1,35 | 0,61 ± 0,02 |
| DT: J48 | 64,81 ± 1,09 | 0,58 ± 0,02 | 60,15 ± 1,42 | 0,59 ± 0,02 |
| DT: Random Forest N=10 | 69,35 ± 0,55 | 0,62 ± 0,02 | 65,93 ± 1,26 | 0,62 ± 0,02 |
| DT: Random Forest N=50 | 69,86 ± 0,48 | 0,66 ± 0,02 | 69,29 ± 0,52 | 0,66 ± 0,03 |
| DT: Random Forest N=100 | 69,98 ± 0,57 | 0,67 ± 0,02 | 69,47 ± 0,36 | 0,66 ± 0,02 |

**Tabla 4.10:** Resultados logrados en cuanto a precisión (%) y AUC, con y sin SMOTE, para VSM utilizando *n-gramas*, respecto al Grado del comentario.

| | Sin SMOTE | | Con SMOTE | |
|---|---|---|---|---|
| **Clasificador** | **Prec.(%)** | **AUC** | **Prec.(%)** | **AUC** |
| KNN K = 1 | 58,69 ± 1,32 | 0,61 ± 0,01 | 47,18 ± 1,53 | 0,61 ± 0,02 |
| KNN K = 2 | 66,78 ± 0,97 | 0,63 ± 0,01 | 53,76 ± 1,42 | 0,64 ± 0,01 |
| KNN K = 3 | 66,73 ± 1,01 | 0,64 ± 0,01 | 46,20 ± 1,75 | 0,65 ± 0,01 |
| KNN K = 4 | 67,15 ± 0,84 | 0,64 ± 0,01 | 45,36 ± 1,39 | 0,65 ± 0,01 |
| KNN K = 5 | 67,90 ± 0,73 | 0,64 ± 0,01 | 42,61 ± 2,19 | 0,66 ± 0,01 |
| BN: Bayes K2 | 48,56 ± 1,65 | 0,62 ± 0,02 | 70,13 ± 0,53 | 0,67 ± 0,02 |
| BN: Bayes TAN | 67,62 ± 0,79 | 0,66 ± 0,02 | 70,70 ± 0,42 | 0,69 ± 0,02 |
| Naïve Bayes | 28,67 ± 1,50 | 0,63 ± 0,02 | 43,79 ± 1,50 | 0,64 ± 0,02 |
| SVM: Polinomial | 72,59 ± 0,56 | 0,57 ± 0,01 | 48,78 ± 2,45 | 0,70 ± 0,02 |
| SVM: Polinomial normal. | 73,24 ± 0,56 | 0,58 ± 0,01 | 57,52 ± 1,65 | 0,67 ± 0,02 |
| SVM: PVK | 71,69 ± 0,44 | 0,54 ± 0,01 | 60,61 ± 1,53 | 0,68 ± 0,02 |
| SVM: RBF | 70,48 ± 0,25 | 0,51 ± 0,00 | 35,56 ± 1,89 | 0,64 ± 0,02 |
| DT: J48 | 64,96 ± 1,84 | 0,63 ± 0,03 | 59,27 ± 1,51 | 0,62 ± 0,02 |
| DT: Random Forest N=10 | 69,18 ± 0,80 | 0,64 ± 0,01 | 66,43 ± 0,99 | 0,64 ± 0,02 |
| DT: Random Forest N=50 | 69,83 ± 0,57 | 0,69 ± 0,01 | 69,53 ± 0,55 | 0,68 ± 0,02 |
| DT: Random Forest N=100 | 69,83 ± 0,59 | 0,70 ± 0,01 | 69,99 ± 0,60 | 0,69 ± 0,02 |

Estos resultados muestran también qué tareas de clasificación son más complicadas de representar. En este sentido, los resultados confirman que lo más sencillo es averiguar si un comentario se centra en la noticia o en otro comentario, mientras que la clasificación más complicada es la correspondiente a establecer el grado de controversia del comentario realizado por el usuario. Por otra parte, también se plantea la duda acerca de la subjetividad del etiquetado, especialmente para la categorización referente a los diferentes niveles de controversia del comentario, tienen una gran dependencia de las opiniones de la persona que etiqueta el conjunto de comentarios.

Sin embargo, los resultados para los 3 tipos diferentes de categorización son sólidos, por lo que nuestro enfoque puede ser usado para clasificar correctamente cada comentario publicado por los usuarios. De esta manera, los resultados también exponen que el empleo de *n-gramas* es más óptimo, con el fin de conformar las características basadas en VSM, que el clásico enfoque basado en palabras. Además, la información sobre los comentarios puede ser empleada para categorizar usuarios respecto al tipo de comentarios que realizan y publican.

### 4.3.2 **Conjunto de usuarios**

Las Tablas 4.11 y 4.12 muestran los resultados obtenidos respecto a la clasificación en términos del tipo de usuario: contribuyente, participativo, comentarista o lector. En esta tarea, el mejor clasificador ha sido *Random Forest* con 100 árboles, con el clásico modelo basado en palabras de VSM y sin equilibrar los datos con SMOTE: una precisión de 80,04 % y un AUC de 0,98. En general, los clasificadores han sido capaces de lograr resultados significativos respecto a AUC y en referencia a la precisión. Los mejores clasificadores han sido los SVM, con la excepción de RBF, y los árboles de decisión (DT), mientras que los clasificadores bayesianos y los basados en instancias han conseguido los peores resultados.

Las Tablas 4.13 y 4.14 muestran los resultados en referencia a la clasificación en términos del nivel de controversia del usuario. En esta tarea, el mejor clasificador ha sido *Random Forest* con un número de árboles igual a 100, con el modelo de VSM basado en palabras y sin balancear los datos con SMOTE: un valor de precisión de 93,59 % y un valor de AUC de 0,99. Generalmente, los clasificadores han alcanzado buenos resultados en términos de AUC, superando los árboles de decisión el valor de 0,95. Respecto a la precisión, los mejores clasificadores han sido los SVM (con la excepción de RBF) y los árboles de decisión, mientras que los clasificadores basados en instancias, el SVM de tipo RBF y Naïve Bayes han alcanzado los peores valores entre el conjunto de clasificadores.

**Tabla 4.11:** Resultados obtenidos en base a la precisión (%) y área bajo la curva ROC, con y sin SMOTE, para VSM usando palabras, respecto a la clasificación Tipo de usuario.

| | Sin SMOTE | | Con SMOTE | |
|---|---|---|---|---|
| **Clasificador** | **Prec.(%)** | **AUC** | **Prec.(%)** | **AUC** |
| KNN K = 1 | 57,80±2,12 | 0,74±0,06 | 56,51±2,43 | 0,79±0,06 |
| KNN K = 2 | 53,55±2,25 | 0,80±0,06 | 50,02±2,74 | 0,84±0,05 |
| KNN K = 3 | 57,94±2,41 | 0,83±0,06 | 56,39±2,41 | 0,87±0,05 |
| KNN K = 4 | 57,55±2,57 | 0,85±0,06 | 55,05±2,58 | 0,88±0,05 |
| KNN K = 5 | 59,71±2,39 | 0,86±0,06 | 57,58±2,77 | 0,89±0,05 |
| BN: Bayes K2 | 67,00±2,22 | 0,97±0,01 | 68,07±2,22 | 0,97±0,01 |
| BN: Bayes TAN | 69,92±2,24 | 0,97±0,01 | 72,97±2,57 | 0,96±0,02 |
| Naïve Bayes | 66,30±2,22 | 0,95±0,03 | 66,14±2,28 | 0,95±0,02 |
| SVM: Polinomial | 68,59±2,03 | 0,90±0,04 | 70,36±2,36 | 0,94±0,03 |
| SVM: Polinomial normal. | 67,47±2,02 | 0,90±0,04 | 67,56±2,43 | 0,93±0,04 |
| SVM: PVK | 71,01±2,00 | 0,91±0,04 | 71,07±2,25 | 0,93±0,03 |
| SVM: RBF | 49,99±0,26 | 0,55±0,04 | 46,58±2,42 | 0,91±0,03 |
| DT: J48 | 77,51±1,89 | 0,86±0,06 | 76,34±2,09 | 0,87±0,06 |
| DT: Random Forest N=10 | 77,39±2,10 | 0,95±0,03 | 76,14±2,35 | 0,95±0,03 |
| DT: Random Forest N=50 | 79,63±2,20 | 0,97±0,02 | 78,79±2,19 | 0,97±0,02 |
| DT: Random Forest N=100 | 80,04±2,22 | 0,98±0,02 | 79,14±2,17 | 0,98±0,02 |

**Tabla 4.12:** Resultados logrados en cuanto a precisión (%) y AUC, con y sin SMOTE, para VSM utilizando *n-gramas*, respecto al Tipo de usuario.

| | Sin SMOTE | | Con SMOTE | |
|---|---|---|---|---|
| **Clasificador** | **Prec.(%)** | **AUC** | **Prec.(%)** | **AUC** |
| KNN K = 1 | 59,84±2,18 | 0,73±0,06 | 56,51±2,43 | 0,79±0,06 |
| KNN K = 2 | 55,40±2,55 | 0,80±0,06 | 50,02±2,74 | 0,84±0,05 |
| KNN K = 3 | 60,33±2,41 | 0,84±0,05 | 56,39±2,41 | 0,87±0,05 |
| KNN K = 4 | 60,45±2,41 | 0,86±0,05 | 55,05±2,58 | 0,88±0,05 |
| KNN K = 5 | 61,01±2,24 | 0,87±0,05 | 57,58±2,77 | 0,89±0,05 |
| BN: Bayes K2 | 67,18±2,11 | 0,97±0,01 | 68,07±2,22 | 0,97±0,01 |
| BN: Bayes TAN | 70,04±2,15 | 0,97±0,02 | 72,97±2,57 | 0,96±0,02 |
| Naïve Bayes | 66,41±2,29 | 0,95±0,03 | 66,14±2,28 | 0,95±0,02 |
| SVM: Polinomial | 68,07±2,09 | 0,89±0,04 | 70,36±2,36 | 0,94±0,03 |
| SVM: Polinomial normal. | 67,98±2,00 | 0,92±0,04 | 67,56±2,43 | 0,93±0,04 |
| SVM: PVK | 73,05±2,01 | 0,93±0,04 | 71,07±2,25 | 0,93±0,03 |
| SVM: RBF | 49,97±0,26 | 0,54±0,03 | 46,58±2,42 | 0,91±0,03 |
| DT: J48 | 77,59±2,00 | 0,86±0,06 | 76,34±2,09 | 0,87±0,06 |
| DT: Random Forest N=10 | 77,55±2,10 | 0,94±0,04 | 76,14±2,35 | 0,95±0,03 |
| DT: Random Forest N=50 | 79,70±2,15 | 0,97±0,02 | 78,79±2,19 | 0,97±0,02 |
| DT: Random Forest N=100 | 79,94±2,18 | 0,98±0,02 | 79,14±2,17 | 0,98±0,02 |

**Tabla 4.13:** Resultados obtenidos en base a la precisión (%) y AUC, con y sin SMOTE, para el modelo VSM utilizando palabras, respecto a la clasificación Grado del usuario.

| | Sin SMOTE | | Con SMOTE | |
|---|---|---|---|---|
| **Clasificador** | **Prec.(%)** | **AUC** | **Prec.(%)** | **AUC** |
| KNN K = 1 | 53,74±2,81 | 0,66±0,03 | 52,72±3,05 | 0,66±0,02 |
| KNN K = 2 | 54,43±2,24 | 0,69±0,03 | 54,84±2,70 | 0,70±0,03 |
| KNN K = 3 | 55,56±2,44 | 0,70±0,03 | 55,25±3,27 | 0,71±0,03 |
| KNN K = 4 | 55,25±2,21 | 0,71±0,03 | 54,95±3,00 | 0,72±0,03 |
| KNN K = 5 | 55,70±2,54 | 0,71±0,03 | 55,06±2,93 | 0,72±0,03 |
| BN: Bayes K2 | 82,21±2,37 | 0,93±0,02 | 68,99±2,48 | 0,89±0,02 |
| BN: Bayes TAN | 84,97±1,90 | 0,97±0,01 | 87,10±1,98 | 0,99±0,01 |
| Naïve Bayes | 26,52±2,31 | 0,64±0,05 | 30,33±2,97 | 0,55±0,04 |
| SVM: Polinomial | 70,80±4,43 | 0,75±0,04 | 87,30±2,23 | 0,93±0,02 |
| SVM: Polinomial normal. | 76,31±3,80 | 0,81±0,04 | 86,30±2,34 | 0,91±0,02 |
| SVM: PVK | 76,76±3,46 | 0,83±0,03 | 84,82±2,20 | 0,90±0,02 |
| SVM: RBF | 47,81±0,23 | 0,50±0,00 | 35,67±8,33 | 0,58±0,04 |
| DT: J48 | 91,85±1,59 | 0,96±0,02 | 92,85±1,50 | 0,96±0,01 |
| DT: Random Forest N=10 | 92,25±1,41 | 0,99±0,00 | 92,58±1,53 | 0,99±0,00 |
| DT: Random Forest N=50 | 93,41±1,37 | 0,99±0,00 | 93,42±1,49 | 0,99±0,00 |
| DT: Random Forest N=100 | 93,59±1,40 | 0,99±0,00 | 93,49±1,39 | 0,99±0,00 |

**Tabla 4.14:** Resultados logrados en cuanto a precisión (%) y AUC, con y sin SMOTE, para VSM utilizando *n-gramas*, respecto al Grado del usuario.

| | Sin SMOTE | | Con SMOTE | |
|---|---|---|---|---|
| **Clasificador** | **Prec.(%)** | **AUC** | **Prec.(%)** | **AUC** |
| KNN 1 | 56,31±2,74 | 0,68±0,03 | 56,47±3,11 | 0,69±0,03 |
| KNN 2 | 57,51±2,27 | 0,73±0,03 | 57,98±2,71 | 0,73±0,03 |
| KNN 3 | 57,86±2,62 | 0,75±0,03 | 57,41±3,20 | 0,74±0,03 |
| KNN 4 | 58,88±2,40 | 0,76±0,03 | 56,59±3,05 | 0,75±0,03 |
| KNN 5 | 58,91±2,58 | 0,76±0,03 | 56,88±3,23 | 0,75±0,03 |
| Bayes K2 | 82,21±2,37 | 0,93±0,02 | 70,01±2,40 | 0,89±0,02 |
| Bayes TAN | 84,97±1,90 | 0,97±0,01 | 87,40±1,98 | 0,99±0,01 |
| Naïve Bayes | 26,29±2,39 | 0,64±0,05 | 29,18±2,91 | 0,54±0,04 |
| SVM: Polinomial | 70,88±4,45 | 0,76±0,04 | 87,20±2,24 | 0,92±0,02 |
| SVM: Polinomial normal. | 84,00±2,75 | 0,89±0,02 | 90,32±1,71 | 0,95±0,01 |
| SVM: PVK | 84,80±2,21 | 0,90±0,02 | 88,16±1,84 | 0,93±0,01 |
| SVM: RBF | 47,81±0,23 | 0,50±0,00 | 44,94±5,57 | 0,58±0,04 |
| DT: J48 | 91,91±1,59 | 0,96±0,01 | 92,91±1,51 | 0,96±0,01 |
| DT: Random Forest N=10 | 92,30±1,59 | 0,99±0,00 | 92,57±1,49 | 0,99±0,01 |
| DT: Random Forest N=50 | 93,35±1,47 | 0,99±0,00 | 93,40±1,42 | 0,99±0,00 |
| DT: Random Forest N=100 | 93,55±1,46 | 0,99±0,00 | 93,55±1,43 | 0,99±0,00 |

Los resultados plantean varias conclusiones, respecto a las etapas de procesamiento, que son adecuadas para cada tipo de clasificación. En primer lugar, el empleo de SMOTE no aporta diferencias significativas en ninguna de las tareas de clasificación. Aunque las clases están desequilibradas, más en la clasificación perteneciente al tipo de usuario que al grado de cada usuario, el SMOTE no es capaz de mejorar los resultados mediante el sobre-muestreo de las clases. En segundo lugar, el empleo *n-gramas* como términos para la mayoría de los clasificadores ofrece mejores resultados clasificatorios que el empleo de palabras como términos para el modelo VSM. La única excepción es *Random Forest* para las 2 clasificaciones, tanto el tipo de usuario como el grado del usuario, que usando *n-gramas* ha obtenido resultados más bajos que el modelo VSM basado en palabras.

Esta mejora utilizando *n-gramas* es consistente con el tipo de datos que hemos aplicado al modelo VSM, los cuales son los comentarios del usuario, las etiquetas de las noticias respecto a las características relacionadas con los comentarios y el texto del *avatar* del perfil del usuario. Normalmente, estas características son de longitud corta y por lo tanto, la cantidad de atributos que el modelado de *n-gramas* obtiene es mayor que cuando, únicamente, se usan palabras. Acorde a nuestros resultados, estos atributos añadidos son valiosos para las labores de clasificación.

Además, si comparamos estos resultados con los obtenidos anteriormente categorizando solamente los comentarios, se destaca que la clasificación de usuarios es más sencilla que la categorización de comentarios. En particular, en la clasificación de comentarios hemos obtenido resultados sólidos pero dicha categorización de comentarios respecto al nivel de polemicidad (grado del comentario) fue la más compleja entre las diferentes clasificaciones. Por el contrario, el categorizar usuarios en niveles de controversia es una tarea más sencilla. Suponemos que la razón es que el sistema emplea un conjunto de comentarios para generar el nivel de controversia global del usuario, permitiendo a los modelos inferir la tendencia de un usuario.

## 4.4 **Sumario**

Este capítulo es la continuidad del capítulo 3, en el cual hemos pre-procesado, modelado y adaptado los conjuntos de datos extraídos del sitio web social *Menéame*, para categorizar tanto los comentarios como los usuarios con sus respectivos comentarios y obtener los resultados pertinentes en esta fase de la investigación.

En primer lugar, hemos revisado las clases de algoritmos pertenecientes al aprendizaje automático supervisado que hemos utilizado a la hora de categorizar, tanto el conjunto de comentarios como el conjunto de usuarios, en diferentes clasificaciones dependiendo de las clases a las que pertenezcan, gracias al proceso de etiquetado realizado previamente.

A continuación, hemos llevado a cabo la validación empírica desarrollada en nuestra metodología, con el fin de refutar la hipótesis planteada al principio de esta investigación. Para ello, hemos aplicado una serie de técnicas y enfoques: (i) la validación cruzada, (ii) el *Information Gain* y (iii) el SMOTE, así como los diferentes algoritmos supervisados con distintas configuraciones para obtener una serie de resultados con mayor o menor alcance. Éstos dependen de las métricas con las que se han evaluado: (i) el TPR, (ii) el FPR, (iii) la precisión y (iv) la AUC. En este apartado, primeramente se han obtenido los resultados pertenecientes a los comentarios. Seguidamente, y siguiendo la misma metodología y validación empírica, los resultados del conjunto de usuarios con sus respectivos comentarios.

Por último, hemos presentado los resultados obtenidos por la categorización de ambos conjuntos mediante el empleo de tablas explicativas. Una vez mostradas y explicadas todas las tablas y sus diferentes configuraciones, hemos procedido a discutir los resultados logrados, mostrando los mejores valores obtenidos para cada tipo de clasificación y esgrimiendo una serie de conclusiones al respecto.

*«No hay talento más valioso que el de no usar dos palabras cuando basta una.»*

Thomas Jefferson (1743-1826)

CAPÍTULO

# 5

# Reducción del esfuerzo de etiquetado

En el capítulo 4, hemos propuesto un enfoque capaz de categorizar automáticamente comentarios y usuarios en sitios web sociales, empleando algoritmos de aprendizaje automático supervisado. No obstante, el aprendizaje supervisado requiere un gran número de datos etiquetados para cada una de las clases (por ejemplo, comentario normal o polémico). Por lo tanto, el etiquetar esta cantidad de información se convierte en una tarea ardua y tediosa, como ocurre en la minería de datos o la minería web, por ejemplo. Para generar estos datos, se debe realizar un costoso proceso de análisis, durante el cual algunos de los datos pueden evitar el filtrado.

Dentro de la amplia gama de enfoques en el aprendizaje automático, hemos propuesto diversas alternativas para hacer frente a este problema, como el aprendizaje semi-supervisado. Este tipo de aprendizaje es una clase de técnica perteneciente al aprendizaje automático que es especialmente útil cuando existe una cantidad limitada de información para cada una de las clases (Chapelle *et al.*, 2006). Particularmente, la clasificación colectiva (Neville y Jensen, 2003) es un enfoque que emplea una estructura relacional sobre la combinación de los conjuntos de datos, tanto etiquetados como sin etiquetar, para incrementar la precisión de la clasificación. Con estos modelos relacionales, la etiqueta prevista se verá influenciada por las etiquetas de las muestras relacionadas. Las

técnicas, tanto del aprendizaje colectivo como del aprendizaje supervisado, han sido implementadas, satisfactoriamente en diferentes campos de las ciencias de la computación como la clasificación de texto (Neville y Jensen, 2003), la detección de *malware*[1] (Santos *et al.*, 2011) o el filtrado de *spam* (Laorden *et al.*, 2011).

Considerando estos antecedentes, hemos implementado un nuevo enfoque para la categorización de texto basado en técnicas de clasificación colectiva con el fin de optimizar el rendimiento clasificatorio, mediante el filtrado de comentarios polémicos o muy polémicos en busca de usuarios trol. Este método emplea una combinación de características estadísticas, sintácticas y de opinión de los comentarios y una combinación de diversas características de los perfiles públicos de los usuarios y de sus comentarios.

En resumen, nuestras principales contribuciones son:

- Un nuevo método para representar comentarios y usuarios en sitios web sociales.
- Una adaptación del enfoque de aprendizaje colectivo, con el fin de filtrar comentarios y usuarios trol.
- Una validación empírica, la cual muestra que nuestro método puede lograr altas tasas de precisión, minimizando el esfuerzo de etiquetado.

El resto del capítulo está estructurado de la siguiente manera. La sección 5.1 describe el método empleado a la hora de extraer las características de los conjuntos de datos de comentarios y usuarios de *Menéame*. La sección 5.2 hace referencia a los diferentes métodos de aprendizaje semi-supervisado y colectivo utilizados en esta investigación. La sección 5.3 desarrolla la validación empírica para nuestra metodología, en cuanto a la categorización de comentarios y usuarios en sitios web sociales. En la sección 5.4 exponemos a los resultados y las conclusiones obtenidas en este capítulo, en referencia a la categorización mediante aprendizaje semi-supervisado de comentarios y usuarios en el sitio web social de noticias *Menéame*. Por último, la sección 5.5 resume los conceptos expresados a lo largo de este capítulo.

[1] *Malware* es la combinación de las voces inglesas *malicious* (malicioso) y *software*, y se refiere a cualquier programa informático diseñado con la intención de dañar ordenadores, redes o información (Vasudevan, 2007; Idika y Mathur, 2007).

## 5.1 Descripción del método

Hemos categorizado tanto los comentarios como los usuarios en 2 clases, ya que la implementación del aprendizaje colectivo utilizada, únicamente nos permite el uso de una clasificación binaria. Por ello, hemos clasificado ambos conjuntos de datos en *Normal* y *Polémico*. La etiqueta de *Normal*, no plantea ninguna controversia tanto del comentario como del usuario en cuestión, mientras que *Polémico*, es un comentario que busca, a propósito, la polémica con un tono perjudicial, generalmente realizado por un usuario trol.

### 5.1.1 Características extraídas

Esta sección expone qué características hemos extraído tanto del conjunto de comentarios como del conjunto de usuarios. Estas características son las mismas que las descritas anteriormente en las secciones 3.3.1 y 3.3.2 y que a continuación vamos a resumir brevemente.

#### 5.1.1.1 Conjunto de comentarios

Esta sección describe cuáles son las características que hemos extraído del conjunto de comentarios. Hemos dividido estas características en 3 categorías diferentes: (i) características estadísticas, (ii) características sintácticas y (iii) características de opinión:

- **Características estadísticas.**
  - *Cuerpo del comentario.* Para representar los comentarios hemos usado el modelo VSM (Baeza-Yates y Ribeiro-Neto, 1999). Hemos utilizado el esquema de ponderación denominado frecuencia del término-frecuencia inversa del documento (TF-IDF, de su denominación inglesa *Term Frequency-Inverse Document Frequency*) (Salton y McGill, 1983). Hemos empleado 2 alternativas diferentes: utilizando la palabra como término y el enfoque de *n-grama* como términos para ponderar.
  - *Número de referencias al comentario.* Indica el número de veces que el comentario ha sido referenciado en otros comentarios de la misma noticia o historia.
  - *Número de referencias desde el comentario.* Mide el número de referencias del comentario a otros comentarios dentro de la misma noticia.

  - *Número de comentario.* Indica la antigüedad del comentario en la noticia. Cuanto menor sea el número del comentario, mayor será la antigüedad del mismo en la noticia.
  - *Similitud del comentario con el resumen de la noticia.* Hemos empleado la similitud del enfoque VSM del comentario con el modelo del resumen de la noticia. En particular, hemos usado la similitud del coseno (Tata y Patel, 2007).
  - *Número de coincidencias entre las palabras del comentario y las etiquetas de la noticia.* Hemos contado el número de palabras que aparecen en el comentario y que aparecen, a su vez, como etiquetas de la historia.
  - *Número de enlaces en el cuerpo del comentario.* Hemos contabilizado el número de enlaces (imágenes, vídeos, páginas web, etc.) dentro del cuerpo del comentario.

- **Características sintácticas.** Hemos realizado un etiquetado de POS (acrónimo de los términos ingleses *Part of Speech*) haciendo uso de la herramienta de código abierto *Freeling*[1] para el análisis lingüístico. Han sido utilizadas las siguientes características:

  - Número de adjetivos en el cuerpo del comentario.
  - Número de números en el cuerpo del comentario.
  - Número de fechas en el cuerpo del comentario.
  - Número de adverbios en el cuerpo del comentario.
  - Número de nombres en el cuerpo del comentario.
  - Número de conjunciones en el cuerpo del comentario.
  - Número de pronombres en el cuerpo del comentario.
  - Número de signos de puntuación en el cuerpo del comentario.
  - Número de interjecciones en el cuerpo del comentario.
  - Número de determinantes en el cuerpo del comentario.
  - Número de abreviaturas en el cuerpo del comentario.
  - Número de preposiciones en el cuerpo del comentario.
  - Número de verbos en el cuerpo del comentario.

[1] Disponible en `http://www.lsi.upc.edu/~nlp/freeling`

- **Características de opinión.**

  - *Número de palabras positivas y negativas.* Hemos empleado un léxico de opinión externo[1]. Debido a que los términos en este léxico están escritos en Inglés y *Menéame* está desarrollado en castellano, hemos traducido este léxico al castellano.
  - *Número de votos.* Es el número de votos positivos del comentario.
  - *Karma.* Hemos recuperado el *karma* computado por el sitio web social *Menéame*.

#### 5.1.1.2 Conjunto de usuarios

En esta sección describimos las características que hemos extraído de los perfiles de los usuarios y sus comentarios. Hemos dividido estas características en 2 categorías diferentes: (i) referentes al perfil y (ii) referentes al comentario:

- **Características referentes al perfil.** Varias características han sido reunidas del perfil público de los usuarios. En particular, las siguientes características:

  - Fecha de registro.
  - *Karma*.
  - *Ranking*.
  - Número de noticias enviadas.
  - Número de noticias publicadas.
  - Entropía.
  - Número de comentarios.
  - Número de notas.
  - Número de votos.
  - Texto del *avatar*.

- **Características referentes a los comentarios.** También hemos analizado los comentarios empleando la categorización, anteriormente realizada, para contabilizar el número de comentarios en cada categoría:

[1] Disponible en: `http://www.cs.uic.edu/~liub/FBS/opinion-lexicon-English.rar`

- *Enfoque del comentario.* El comentario puede enfocarse a la noticia o a otros comentarios.
- *Tipo de comentario.* Se puede diferenciar entre aporte, irrelevante u opinión a la noticia o a otro comentario.
- *Grado del comentario.* Puede ser normal, polémico, muy polémico o gracioso.

El enfoque usado utiliza diferentes características sintácticas, estadísticas y de opinión para representar los comentarios, las cuales han sido explicadas en la sección 3.3.1 y que repasamos a continuación:

- *Características estadísticas*.
  - Cuerpo del comentario.
  - Número de referencias al comentario.
  - Número de referencias desde el comentario.
  - Número de comentario en la noticia.
  - Similitud del comentario respecto al resumen de la noticia.
  - Número de coincidencias entre las palabras del comentario y las etiquetas de la noticia.
  - Número de enlaces en el cuerpo del comentario.
- *Características sintácticas*.
  - Número de adjetivos en el cuerpo del comentario.
  - Número de números en el cuerpo del comentario.
  - Número de fechas en el cuerpo del comentario.
  - Número de adverbios en el cuerpo del comentario.
  - Número de nombres en el cuerpo del comentario.
  - Número de conjunciones en el cuerpo del comentario.
  - Número de pronombres en el cuerpo del comentario.
  - Número de signos de puntuación en el cuerpo del comentario.
  - Número de interjecciones en el cuerpo del comentario.
  - Número de determinantes en el cuerpo del comentario.
  - Número de abreviaturas en el cuerpo del comentario.
  - Número de preposiciones en el cuerpo del comentario.
  - Número de verbos en el cuerpo del comentario.

- *Características de opinión*.
  - Número de palabras positivas y negativas.
  - Número de votos.
  - *Karma*.

Por lo tanto, hemos categorizado los comentarios de los usuarios empleando el enfoque realizado con anterioridad, el cual nos permite generar 9 nuevas características que contabilizan el número de comentarios en estas categorías:

- Número de comentarios referentes a la noticia.
- Número de comentarios referentes a otros comentarios.
- Número de comentarios del tipo aporte.
- Número de comentarios irrelevantes.
- Número de comentarios de opinión.
- Número de comentarios normales.
- Número de comentarios polémicos.
- Número de comentarios muy polémicos.
- Número de comentarios graciosos.

## 5.2 La clasificación colectiva

La clasificación colectiva es un problema de optimización combinatoria, en el cual se nos ofrece un conjunto de usuarios (con sus comentarios), o nodos, $\mathscr{U} = \{u_1, u_2, ..., u_n\}$ y una función de vecindad $V$, donde $V_i \subseteq \mathscr{U} \setminus \{\mathscr{U}_i\}$, que describe la estructura de la red subyacente (Namata *et al.*, 2009). Siendo $\mathscr{U}$ una colección aleatoria de usuarios, se divide en dos conjuntos $\mathscr{X}$ e $\mathscr{Y}$, donde $\mathscr{X}$ corresponde a los usuarios para los que conocemos los valores correctos e $\mathscr{Y}$ son los usuarios cuyos valores necesitan ser determinados. Por lo tanto, la tarea consiste en etiquetar los nodos $\mathscr{Y}_i \in \mathscr{Y}$ con la menor cantidad de etiquetas posibles $\mathscr{E} = \{e_1, e_2, ..., e_q\}$.

Para ello, hemos empleado la herramienta WEKA (acrónimo del inglés *Waikato Environment for Knowledge Analysis*) (Garner *et al.*, 1995), junto con el aprendizaje semi-supervisado y el complemento para la clasificación colectiva[1].

[1]Disponible en: `http://www.cms.waikato.ac.nz/~fracpete/projects/collective-classification`

Durante esta sección, revisamos los algoritmos colectivos utilizados posteriormente en la evaluación empírica.

### 5.2.1 *Collective IBK*

Internamente, el clasificador *Collective IBK* aplica una implementación del clasificador KNN (*K*-vecinos más cercanos), el clásico algoritmo IBK de WEKA, para determinar las mejores *k* instancias en el conjunto de entrenamiento y entonces generar, para cada una de las instancias del conjunto de entrenamiento, un vecindario consistente en *k* instancias tanto del conjunto de entrenamiento como del conjunto de prueba (o bien se usa una búsqueda ingenua sobre el conjunto completo de instancias o un árbol *k*-dimensional para determinar los vecindarios).

Todos los vecinos pertenecientes a un vecindario son ordenados acorde a su distancia respecto a la instancia de prueba a la cual pertenecen. Los vecindarios son ordenados en base a su *rango*. La clase de etiqueta es determinada por mayoría de votos o, en caso de empate, por la primera clase de todas las instancias de prueba no etiquetadas con el valor más alto de *rango*. Esto se lleva a cabo hasta que no quedan más instancias de prueba sin etiquetar. La clasificación acaba mediante la devolución del tipo de instancia, dado por la clase de etiqueta, la cual está a punto de ser clasificada.

### 5.2.2 *CollectiveForest*

En el clasificador *CollectiveForest*, la implementación de WEKA de *RandomTree* se utiliza como clasificador base para dividir el conjunto de prueba en subconjuntos que contienen el mismo número de elementos. La primera repetición entrena el modelo usando el conjunto de entrenamiento original y genera la distribución para todas las instancias en el conjunto de prueba. Entonces, las mejores instancias son añadidas al conjunto de entrenamiento original (eligiendo el mismo número de instancias en un subconjunto). Las siguientes repeticiones entrenan con el nuevo conjunto de entrenamiento y producen las distribuciones para las instancias restantes en el conjunto de prueba.

### 5.2.3 *CollectiveWoods* y *CollectiveTree*

El clasificador *CollectiveWoods*, en asociación con el algoritmo *CollectiveTree*, opera igual que el clasificador *CollectiveForest* utilizando del algoritmo *RandomTree*. *CollectiveTree* es similar al clasificador original de WEKA *RandomTree*.

Su funcionamiento es el siguiente, el atributo es dividido en una posición, la cual fracciona el actual subconjunto de instancias (tanto las instancias de entrenamiento como las de prueba) en 2 mitades. El proceso termina si alguna de las siguientes condiciones se cumple: (i) únicamente las instancias de entrenamiento están cubiertas (las etiquetas para esas instancias son conocidas), (ii) únicamente se encuentran instancias de prueba en las hojas, en cuyo caso la distribución se toma del nodo padre y (iii) únicamente las instancias del conjunto de entrenamiento son de una clase, en cuyo caso se considera que todas las instancias del conjunto de prueba tienen dicha clase.

Los pesos son sumados de acuerdo a los pesos del conjunto de prueba y más adelante son normalizados para calcular la distribución de la clase de un conjunto completo o de un subconjunto. La distribución de un atributo nominal se corresponde con la suma normalizada de los pesos para cada valor distinto y, para el atributo numérico, se calcula la división binaria de la distribución en base a la mediana. En ese momento, los pesos son sumados para las 2 distribuciones y finalmente, son normalizados.

### 5.2.4 *RandomWoods*

Este clasificador actúa igual que el clásico clasificador *RandomForest* de WEKA pero empleando *CollectiveBagging*, en combinación con el algoritmo *CollectiveTree* en lugar de los algoritmos *Bagging* y *RandomTree*. El clásico *Bagging* es un meta-algoritmo que engloba un grupo de clasificadores de aprendizaje automático para mejorar la estabilidad y la precisión. En este caso, ha sido extendido para que esté disponible para los clasificadores colectivos.

## 5.3 Validación empírica

Esta sección describe cómo hemos validado nuestro enfoque a través de un conjunto de comentarios y posteriormente, de usuarios con sus comentarios. Hemos etiquetado cada comentario y cada usuario en una categoría: el nivel de controversia. Esta categoría se refiere a que un comentario o usuario puede ser *Normal* o *Polémico*. *Normal* significa que no es perjudicial o dañino, usando en

su argumento un tono comedido. Mientras que *Polémico* hace referencia a un comentario, el cual busca crear polémica pero no en un modo exagerado, al igual que el usuario que busca crear polémica y al cual lo consideramos trol.

**Tabla 5.1:** Distribución de ejemplos para la categorización colectiva en el conjunto de comentarios y usuarios.

| Conjunto de datos | Normal | Polémico | Totales |
|---|---|---|---|
| Comentarios | 6.857 | 2.187 | 9.044 |
| Usuarios | 1.997 | 1.362 | 3.359 |

Para este fin, hemos construido un conjunto de datos, tanto de comentarios como de usuarios, como se puede apreciar en la Tabla 5.1, donde tanto los 9.044 comentarios como los 3.359 han sido etiquetados manualmente en su correspondiente categoría perteneciente al grado de polemicidad.

### 5.3.1 Metodología

Cuando el trabajo de creación, procesado y analizado de los conjuntos de datos ha terminado, hemos desarrollado una aplicación para obtener todas las características descritas en la sección 5.1. Hemos implementado 2 procedimientos diferentes para construir el enfoque VSM del cuerpo del comentario y del texto del *avatar*: (i) VSM con palabras como términos y (ii) *n-gramas* con diferentes valores de $n$ ($n = 1$, $n = 2$ y $n = 3$).

Además, hemos suprimido todas las palabras carentes de significado en el texto, denominadas palabras de parada (del inglés *stop-words*)(Wilbur y Sirotkin, 1992) como por ejemplo *a*, *el*, *un*. Para este fin, hemos empleado una lista de palabras de parada externa de términos en castellano[1].

De este modo, hemos realizado los correspondientes archivos ARFF (acrónimo del inglés *Attribute Relation File Format*) (Holmes *et al.*, 1994), para más adelante poder hacer uso de la herramienta WEKA (Garner *et al.*, 1995). En este caso, hemos construido 4 archivos diferentes: (i) 2 archivos pertenecientes al conjunto de comentarios (VSM con palabras y VSM con *n-gramas*) y (ii) otros 2 archivos pertenecientes al conjunto de usuarios (VSM con palabras y VSM con *n-gramas*), todos ellos haciendo referencia a la clasificación propuesta: el nivel de controversia o polemicidad.

[1]La lista de palabras de parada puede descargarse en: `http://paginaspersonales.deusto.es/isantos/resources/stopwords.txt`

Una vez realizado el proceso de aplicación del enfoque VSM a ciertas características de nuestros conjuntos de datos y la creación de los archivos ARFF, la Tabla 5.2 muestra los resultados conseguidos para los diferentes conjuntos de comentarios y usuarios.

**Tabla 5.2:** Número de características para cada conjunto de datos para la clasificación colectiva.

| **Conjunto de datos** | **# Características con palabras** | **# Características con *n-gramas*** |
|---|---|---|
| Comentarios | 13.247 | 34.126 |
| Usuarios | 2.120 | 3.600 |

Posteriormente, hemos evaluado la precisión de nuestro método propuesto. Para este fin, hemos seguido la siguiente metodología para los conjuntos de datos de comentarios y usuarios:

- **Ganancia de Información.** Hemos extraído las características principales para cada uno de los tipos de clasificación empleando la ganancia de información (IG, de los términos ingleses *Information Gain*) (Kent, 1983), un algoritmo que evalúa la relevancia de un atributo midiendo la ganancia de información con respecto a la clase. Empleando esta medida, hemos eliminado cualquier característica en el conjunto de entrenamiento que tenga un valor de IG $= 0$. Como se puede observar en las Tablas 5.3 y 5.4, la aplicación del filtro IG permite eliminar los términos innecesarios para la categorización, logrando una importante reducción de ruido, un aumento de la eficiencia e incluso un ligero crecimiento de la efectividad.

**Tabla 5.3:** Reducción de características en el conjunto de comentarios mediante *Information Gain* en la clasificación colectiva.

| **Conjunto de datos** | **# Características con palabras** | **# Características con *n-gramas*** |
|---|---|---|
| Comentarios sin IG | 13.247 | 34.126 |
| Comentarios con IG | 276 | 920 |
| Reducción (%) | 97,91% | 97,30% |

- **SMOTE.** Los conjuntos de datos no están balanceados en las diferentes clases, debido a la heterogeneidad de los datos extraídos. Para hacer frente

**Tabla 5.4:** Reducción de características en el conjunto de usuarios mediante *Information Gain* en la clasificación colectiva.

| **Conjunto de datos** | **# Características con palabras** | **# Características con *n-gramas*** |
|---|---|---|
| Usuarios sin IG | 2.120 | 3.600 |
| Usuarios con IG | 21 | 21 |
| Reducción (%) | 99,01 % | 99,42 % |

a los datos desbalanceados, hemos aplicado la técnica SMOTE (acrónimo de las voces inglesas *Synthetic Minority Over-sampling TEchnique*) (Chawla *et al.*, 2002), la cual es una combinación de sobre-muestreo de las clases menos pobladas y sub-muestreo de las más pobladas.

- **Validación cruzada.** Este método se aplica generalmente en la evaluación del aprendizaje automático (Bishop, 1995). En nuestros experimentos, hemos utilizado la variable $k = 10$ refiriéndose a los $k$-subconjuntos en los que se ejecutará la validación cruzada. Como consecuencia, nuestro conjunto de datos es dividido 10 veces en 10 conjuntos de datos diferentes tanto de entrenamiento como de prueba. Hemos alternado el número de instancias etiquetadas desde el 10 % hasta el 90 % (10 %, 20 %, 25 %, 33 %, 50 %, 66 %, 75 %, 80 % y 90 %) para cada subconjunto. Realizando esta operación, hemos medido el efecto que tiene la variación del número de instancias, previamente etiquetadas, sobre el comportamiento final de la clasificación colectiva.

- **Aprendizaje del modelo.** Para cada subconjunto de datos, hemos efectuado la etapa de aprendizaje de cada algoritmo utilizando diferentes parámetros, dependiendo del modelo específico. Estos algoritmos usan los parámetros por defecto en la herramienta de aprendizaje automático WEKA (acrónimo del inglés *Waikato Environment for Knowledge*). En particular, hemos usado los siguiente modelos:

  - *Collective IBK*. Hemos experimentado con el valor de $k = 10$.
  - *CollectiveForest*. Donde el valor de los árboles con los que experimentamos es de 100.
  - *CollectiveWoods*. Hemos experimentado con 100 árboles.
  - *RandomWoods*. Hemos realizado experimentos con 100 árboles.

En el caso de *CollectiveForest*, *CollectiveWoods* y *RandomWoods*, hemos aplicado el valor por defecto del clasificador: 100 árboles.

- **Evaluación del modelo.** Hemos medido la tasa de verdaderos positivos (TPR, de las voces inglesas *True Positive rate*) para probar nuestro procedimiento tanto para el conjunto de usuarios como para el conjunto de comentarios; es decir, el número de usuarios o comentarios polémicos correctamente detectados divididos por el número total de usuarios o comentarios polémicos:

$$TPR = \frac{TP}{(TP + FN)} \tag{5.1}$$

donde $TP$ es el número de usuarios o comentarios polémicos correctamente clasificados (verdaderos positivos) y $FN$ es el número de usuarios o comentarios polémicos erróneamente clasificados como usuarios o comentarios normales (falsos negativos). Por otra parte, también hemos tenido en cuenta la tasa de falsos positivos (FPR, del inglés *False Positive Rate*); es decir, el número de usuarios o comentarios normales mal clasificados divididos por el número total de usuarios o comentarios normales:

$$FPR = \frac{FP}{(FP + TN)} \tag{5.2}$$

donde $FP$ es el número de usuarios o comentarios incorrectamente detectados y $TN$ es el número de usuarios o comentarios normales correctamente clasificados. Igualmente, hemos obtenido la precisión; es decir, el número total de instancias correctamente clasificadas dividido por el número total de instancias en todo el conjunto de datos:

$$Precision(\%) = \frac{(TP + TN)}{(TP + FP + FN + TN)} \tag{5.3}$$

Finalmente, hemos recuperado el área bajo la curva ROC (AUC). Esta medida establece la relación entre los falsos negativos y los falsos positivos (Singh *et al.*, 2009). Representando gráficamente el valor de TPR contra el valor de FPR, podemos obtener la curva ROC.

## 5.4 Discusión de los resultados

En nuestros experimentos, hemos examinado diferentes configuraciones de los algoritmos colectivos con diferentes tamaños de conjunto $\mathscr{X}$ de instancias conocidas; variando este último desde el 10 % hasta el 90 % de las instancias utilizadas para el entrenamiento (es decir, las instancias conocidas durante la prueba). En las diferentes figuras, se muestran a continuación los resultados obtenidos para las diversas medidas seleccionadas para probar nuestro modelo (precisión, TPR, FPR y AUC).

Por otra parte, y para evaluar la contribución de la clasificación colectiva a la categorización tanto de usuarios como de comentarios, hemos comparado las capacidades de filtrado de nuestro método con algunos de los algoritmos del aprendizaje automático supervisado más empleados. Específicamente, hemos utilizado los siguientes modelos:

- *Redes bayesianas (BN, del inglés Bayesian Networks)*. Hemos utilizado diferentes algoritmos de aprendizaje estructural: K2 (Cooper y Herskovits, 1991b) y *Tree Augmented Naïve* (TAN) (Geiger *et al.*, 1997). Por otra parte, también hemos realizado experimentos con un clasificador *Naïve Bayes* (Bishop, 1995).

- *Máquinas de soporte vectorial (SVM, del inglés Support Vector Machines)*. Hemos elaborado experimentos con un núcleo polinomial (del inglés *polynomial kernel*) (Amari y Wu, 1999), un núcleo polinomial normalizado (del inglés *normalized polynomial kernel*) (Maji *et al.*, 2008), un núcleo *Pearson VII* (Üstün *et al.*, 2007) y un núcleo basado en funciones de base radial (RBF, del inglés *radial basis function*) (Cho *et al.*, 2008).

- *K-vecinos más cercanos (KNN, del inglés K-nearest neighbour)*. Hemos realizado experimentos con el valor de $K = 10$.

- *Árboles de decisión (DT, del inglés Decision Trees)*. Hemos efectuado experimentos con J48 (la implementación de WEKA (Garner *et al.*, 1995) del algoritmo *C4.5* (Quinlan, 1993)) y *Random Forest*, un conjunto de árboles de decisión construidos al azar. Particularmente, hemos probado *Random Forest* con $N = 100$ como número de árboles.

**Tabla 5.5:** Resultados en términos de precisión, tasa de verdaderos positivos (TPR), tasa de falsos positivos (FPR) y área bajo la curva ROC (AUC) sobre el nivel de controversia del comentario, para VSM empleando palabras.

| Clasificador | Precisión (%) | TPR | FPR | AUC |
|---|---|---|---|---|
| KNN K = 10 | 77,68 ± 0,46 | 0,14 ± 0,01 | 0,02 ± 0,01 | 0,64 ± 0,02 |
| BN: Bayes K2 | 76,71 ± 1,21 | 0,15 ± 0,02 | 0,04 ± 0,01 | 0,61 ± 0,03 |
| BN: Bayes TAN | 77,37 ± 0,71 | 0,13 ± 0,03 | 0,02 ± 0,00 | 0,61 ± 0,03 |
| Naïve Bayes | 47,93 ± 2,68 | 0,75 ± 0,04 | 0,61 ± 0,04 | 0,64 ± 0,02 |
| SVM: Polinomial | 78,73 ± 0,73 | 0,17 ± 0,02 | 0,02 ± 0,00 | 0,58 ± 0,01 |
| SVM: Polinomial normal. | 79,49 ± 0,81 | 0,22 ± 0,03 | 0,02 ± 0,00 | 0,60 ± 0,01 |
| SVM: PVK | 79,08 ± 0,81 | 0,19 ± 0,03 | 0,02 ± 0,00 | 0,59 ± 0,01 |
| SVM: RBF | 75,82 ± 0,05 | 0,00 ± 0,00 | 0,00 ± 0,00 | 0,50 ± 0,00 |
| DT: J48 | 70,95 ± 1,19 | 0,29 ± 0,02 | 0,16 ± 0,01 | 0,57 ± 0,03 |
| DT: Random Forest N=100 | 77,31 ± 0,96 | 0,17 ± 0,04 | 0,03 ± 0,01 | 0,69 ± 0,03 |

**Tabla 5.6:** Resultados en términos de precisión, tasa de verdaderos positivos (TPR), tasa de falsos positivos (FPR) y área bajo la curva ROC (AUC) sobre el nivel de controversia del comentario, para VSM empleando *n-gramas*.

| Clasificador | Precisión (%) | TPR | FPR | AUC |
|---|---|---|---|---|
| KNN K = 10 | 75,82 ± 0,77 | 0,13 ± 0,03 | 0,04 ± 0,01 | 0,62 ± 0,02 |
| BN: Bayes K2 | 76,68 ± 1,10 | 0,15 ± 0,03 | 0,04 ± 0,01 | 0,62 ± 0,03 |
| BN: Bayes TAN | 77,28 ± 0,91 | 0,13 ± 0,03 | 0,02 ± 0,00 | 0,62 ± 0,03 |
| Naïve Bayes | 45,72 ± 1,25 | 0,81 ± 0,02 | 0,65 ± 0,02 | 0,63 ± 0,02 |
| SVM: Polinomial | 78,48 ± 0,72 | 0,17 ± 0,03 | 0,02 ± 0,01 | 0,58 ± 0,01 |
| SVM: Polinomial normal. | 79,41 ± 0,89 | 0,22 ± 0,03 | 0,02 ± 0,00 | 0,60 ± 0,02 |
| SVM: PVK | 77,72 ± 0,58 | 0,13 ± 0,02 | 0,02 ± 0,01 | 0,56 ± 0,01 |
| SVM: RBF | 76,18 ± 0,16 | 0,02 ± 0,01 | 0,00 ± 0,00 | 0,51 ± 0,00 |
| DT: J48 | 71,03 ± 1,76 | 0,32 ± 0,04 | 0,17 ± 0,02 | 0,59 ± 0,03 |
| DT: Random Forest N=100 | 77,20 ± 0,75 | 0,16 ± 0,03 | 0,03 ± 0,01 | 0,70 ± 0,03 |

### 5.4.1 Conjunto de comentarios

La Tabla 5.5 muestra los resultados con el enfoque VSM empleando palabras y el aprendizaje supervisado, cuando no es aplicado el SMOTE, la Tabla 5.6 muestra los resultados cuando se emplean los *n-gramas* como términos en el enfoque VSM y el aprendizaje supervisado, cuando no se aplica SMOTE, la Tabla 5.7 muestra los resultados con las palabras como términos para el enfoque VSM y el aprendizaje supervisado, cuando el SMOTE es aplicado y la Tabla 5.8 muestra los resultados con *n-gramas* como términos como enfoque VSM a emplear y el aprendizaje supervisado, aplicando la técnica de SMOTE.

Estas 4 tablas contienen los resultados del aprendizaje supervisado sobre el

**Tabla 5.7:** Resultados en términos de precisión, TPR, FPR y AUC para el nivel de controversia de los comentarios, para el enfoque VSM basado en palabras, aplicando SMOTE.

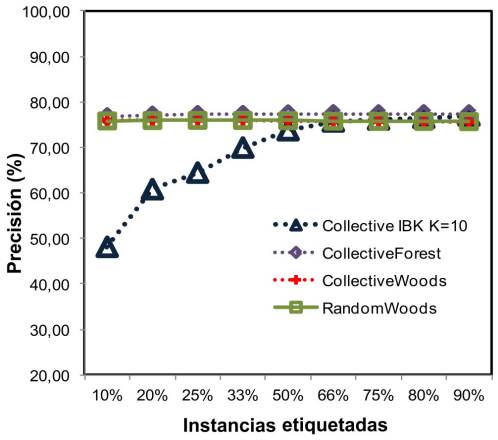
| Clasificador | Precisión (%) | TPR | FPR | AUC |
|---|---|---|---|---|
| KNN K = 10 | 67,14 ± 1,92 | 0,50 ± 0,04 | 0,27 ± 0,02 | 0,66 ± 0,02 |
| BN: Bayes K2 | 75,93 ± 0,65 | 0,04 ± 0,01 | 0,01 ± 0,01 | 0,64 ± 0,03 |
| BN: Bayes TAN | 76,64 ± 0,36 | 0,05 ± 0,01 | 0,01 ± 0,01 | 0,64 ± 0,03 |
| Naïve Bayes | 74,13 ± 3,74 | 0,20 ± 0,11 | 0,09 ± 0,08 | 0,62 ± 0,03 |
| SVM: Polinomial | 68,35 ± 2,06 | 0,59 ± 0,05 | 0,29 ± 0,03 | 0,65 ± 0,03 |
| SVM: Polinomial normal. | 69,53 ± 1,55 | 0,53 ± 0,03 | 0,25 ± 0,02 | 0,64 ± 0,02 |
| SVM: PVK | 69,54 ± 1,33 | 0,52 ± 0,04 | 0,25 ± 0,02 | 0,63 ± 0,02 |
| SVM: RBF | 68,34 ± 3,33 | 0,44 ± 0,03 | 0,24 ± 0,05 | 0,60 ± 0,02 |
| DT: J48 | 71,72 ± 2,06 | 0,31 ± 0,04 | 0,15 ± 0,02 | 0,60 ± 0,04 |
| DT: Random Forest N=100 | 77,08 ± 0,94 | 0,18 ± 0,04 | 0,04 ± 0,01 | 0,67 ± 0,03 |

**Tabla 5.8:** Resultados en términos de precisión, TPR, FPR y AUC para el nivel de controversia de los comentarios, para el enfoque VSM basado en *n-gramas*, aplicando SMOTE.

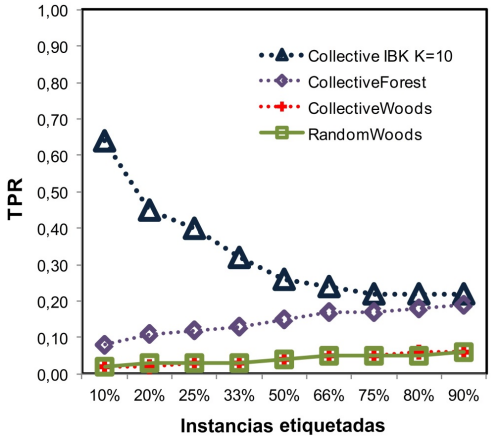
| Clasificador | Precisión (%) | TPR | FPR | AUC |
|---|---|---|---|---|
| KNN K = 10 | 57,32 ± 2,13 | 0,61 ± 0,05 | 0,44 ± 0,03 | 0,63 ± 0,03 |
| BN: Bayes K2 | 75,60 ± 0,74 | 0,06 ± 0,02 | 0,02 ± 0,01 | 0,65 ± 0,02 |
| BN: Bayes TAN | 76,34 ± 0,43 | 0,06 ± 0,02 | 0,01 ± 0,00 | 0,65 ± 0,02 |
| Naïve Bayes | 53,81 ± 1,78 | 0,62 ± 0,02 | 0,49 ± 0,02 | 0,59 ± 0,02 |
| SVM: Polinomial | 60,84 ± 1,38 | 0,74 ± 0,04 | 0,43 ± 0,01 | 0,65 ± 0,02 |
| SVM: Polinomial normal. | 70,72 ± 1,56 | 0,54 ± 0,05 | 0,24 ± 0,02 | 0,65 ± 0,02 |
| SVM: PVK | 70,83 ± 1,86 | 0,49 ± 0,05 | 0,22 ± 0,02 | 0,63 ± 0,03 |
| SVM: RBF | 53,42 ± 2,98 | 0,74 ± 0,03 | 0,53 ± 0,04 | 0,60 ± 0,03 |
| DT: J48 | 71,04 ± 1,54 | 0,35 ± 0,04 | 0,17 ± 0,02 | 0,61 ± 0,02 |
| DT: Random Forest N = 100 | 76,88 ± 1,30 | 0,19 ± 0,04 | 0,05 ± 0,01 | 0,68 ± 0,03 |

enfoque VSM basado en palabras como términos, con y sin la técnica SMOTE aplicada y el enfoque VSM basado en *n-gramas* como términos, con y sin haber empleado el método SMOTE.

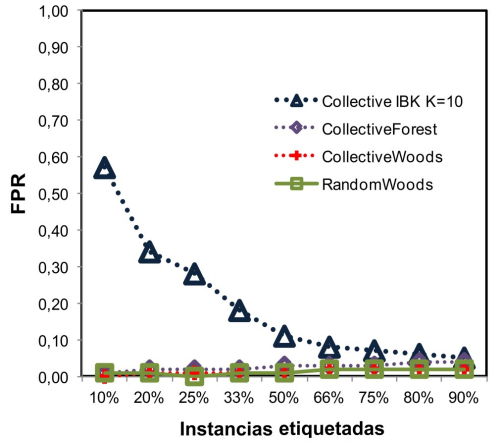
La Figura 5.1 muestra los resultados con el enfoque VSM generado usando palabras como términos, aprendizaje colectivo y sin aplicar la técnica de SMOTE; la Figura 5.2 expone los resultados con el modelo VSM implementado con *n-gramas*, aprendizaje semi-supervisado colectivo y sin SMOTE; la Figura 5.3 muestra los resultados con el modelo VSM generado mediante palabras, el aprendizaje colectivo y aplicando SMOTE y por último, la Figura 5.4 expone los resultados con el enfoque generado con *n-gramas* como términos en el modelo, aprendizaje semi-supervisado colectivo y aplicando la técnica de SMOTE.



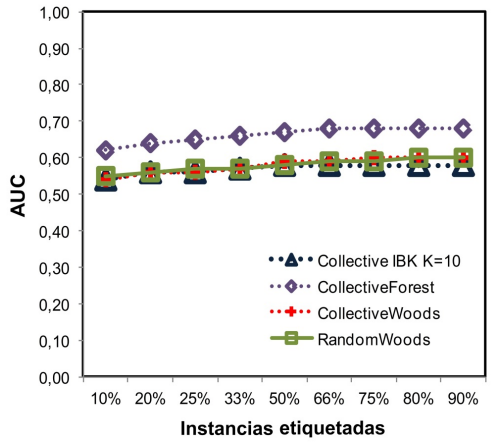

(a) **Resultados de precisión.** El clasificador *CollectiveForest* con un 90 % etiquetado del conjunto de datos y alcanzando un valor de 77,38 % es el mejor clasificador en esta medida.




(b) **Resultados de TPR.** El mejor clasificador es *Collective IBK*, con un valor de precisión de 64 % únicamente etiquetando el 10 % del todo el conjunto de datos.

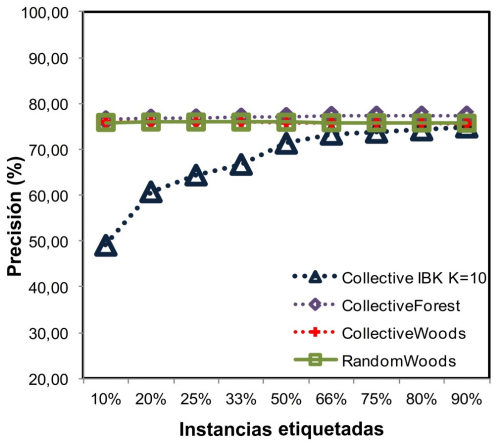



(c) **Resultados de FPR.** La mayoría de los clasificadores, excepto *Collective IBK*, ofrecen valores notables alrededor de 0,1. *Collective-Woods*, obtiene valores cercanos a 0 y con un esfuerzo de etiquetado del 10 % sobre todo el conjunto de datos.

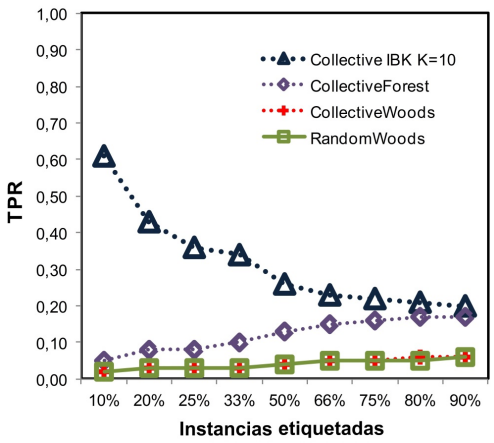



(d) **Resultados de AUC.** Los resultados respecto a AUC obtenidos por todos los clasificadores son muy similares entre ellos. Pero *CollectiveForest* ofrece un resultado de 0,68, con el 66 % de las instancias etiquetadas del conjunto de datos.

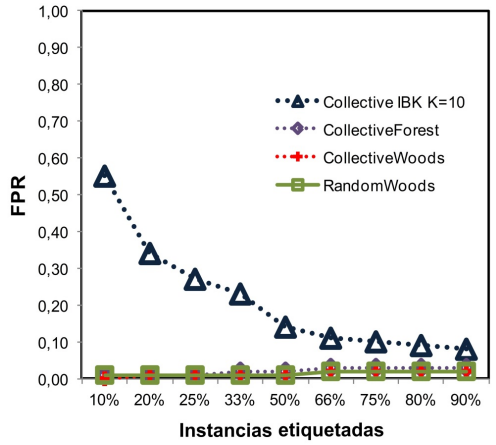
**Figura 5.1:** Resultados de nuestra clasificación colectiva para la categorización de comentarios empleando el modelo de VSM basado en palabras. En resumen, el clasificador *CollectiveForest* es el mejor del conjunto de clasificadores para este enfoque.



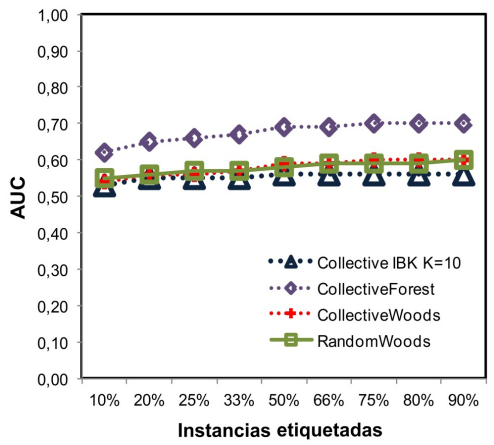

(a) **Resultados de precisión.** El clasificador *CollectiveForest* empleando únicamente el 10 % de las instancias etiquetadas del conjunto de comentarios, logra alcanzar un valor de precisión del 76,40 %.




(b) **Resultados de TPR.** *Collective IBK* es el mejor de los clasificadores obteniendo un valor de TPR de 0,61 solamente etiquetando un 10 % del conjunto de datos.

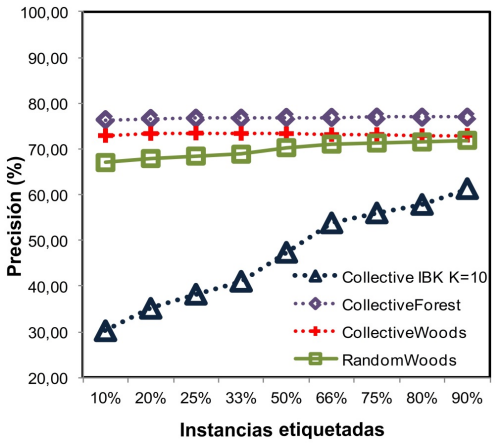



(c) **Resultados de FPR.** El clasificador *CollectiveWoods* consigue resultados cercanos al valor 0, con un 10 % de esfuerzo de etiquetado sobre el conjunto de comentarios.

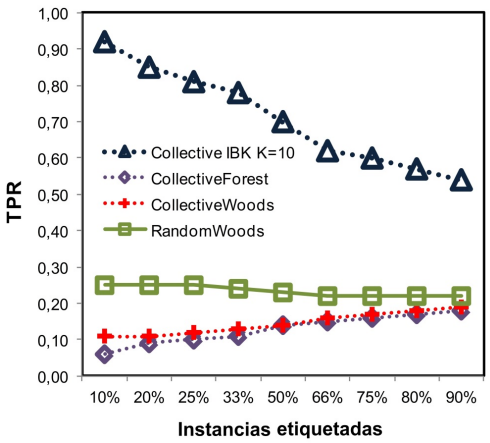



(d) **Resultados de AUC.** Respecto a los resultados obtenidos en AUC, el mejor algoritmo es *CollectiveForest*. Este clasificador logra un valor de 0,70 etiquetando solamente el 75 % del conjunto de datos formado por comentarios.

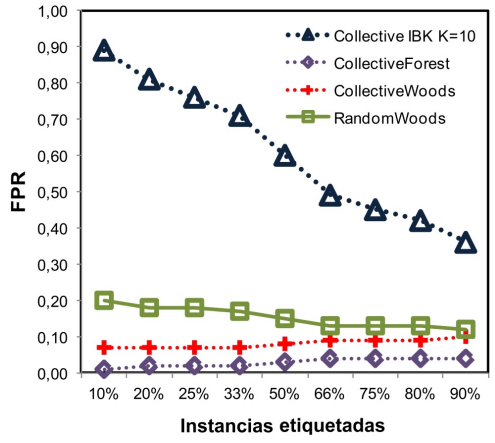
**Figura 5.2:** Resultados de nuestra clasificación colectiva para la categorización de comentarios empleando el modelo de VSM basado en *n-gramas*. Resumiendo, el algoritmo *CollectiveForest* es el clasificador que mejores resultados obtiene de entre todos los ejecutados.



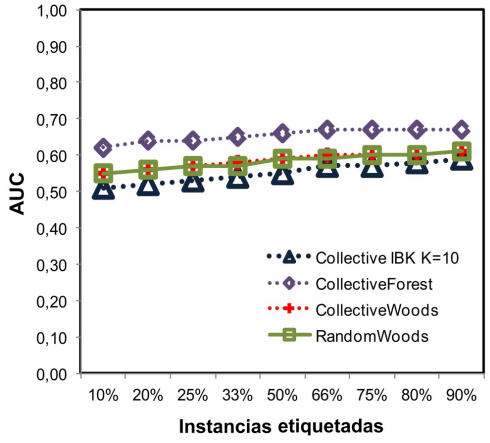

(a) **Resultados de precisión.** El algoritmo *CollectiveForest* obtiene el mayor valor de precisión: 76,94 % etiquetando el 75 % de las instancias del conjunto de datos de comentarios.




(b) **Resultados de TPR.** El clasificador *Collective IBK* logra resultados notables para un número mínimo de instancias etiquetadas, con un valor de 0,92 empleando solamente el 10 % del conjunto de datos.

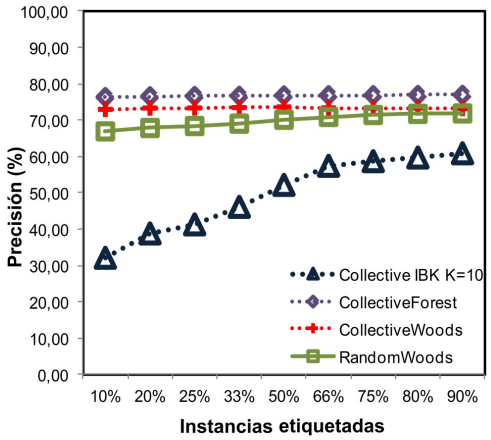



(c) **Resultados de FPR.** Tanto el clasificador *CollectiveForest* como el clasificador *CollectiveWoods* obtienen resultados bastante bajos para esta clasificación, con resultado cercanos a un valor de 0,1.

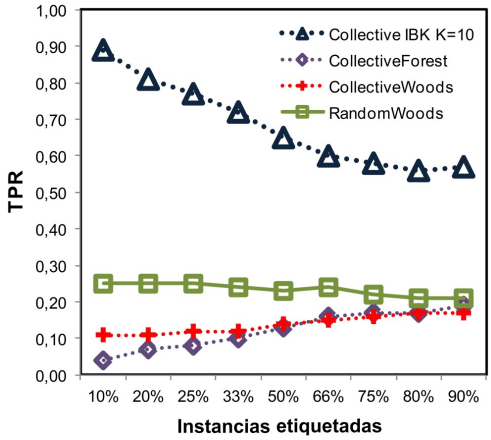



(d) **Resultados de AUC.** El clasificador *CollectiveForest*, etiquetando un 66 % de las instancias sobre el total del conjunto de comentarios, logra alcanzar un valor de 0,67.

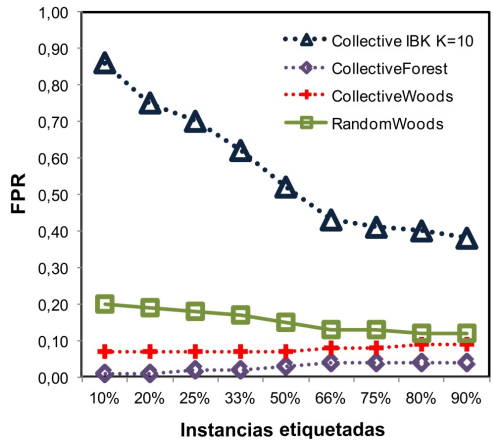
**Figura 5.3:** Resultados realizados con las características del enfoque VSM empleando palabras y aplicando la técnica de SMOTE. Consecuentemente, el clasificador *CollectiveForest* logra los resultados más destacados en diferentes medidas como la precisión, el FPR y la AUC.



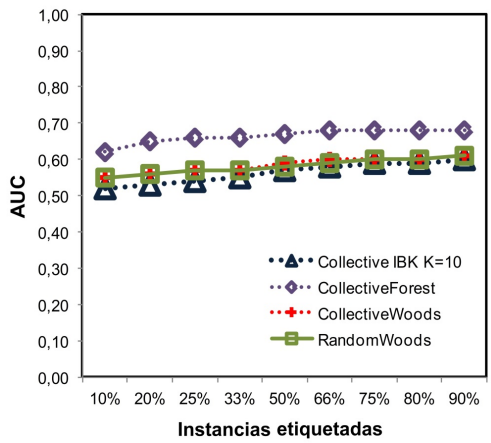

(a) **Resultados de precisión.** El mejor resultado es el logrado por el clasificador *CollectiveForest* etiquetando el 90 % del conjunto de datos: un valor de 76,93 % de precisión.




(b) **Resultados de TPR.** El clasificador *Collective IBK* logra obtener el mejor resultado de entre todos los demás clasificadores: un valor de 0,89, etiquetando exclusivamente el 10 % del total del conjunto de datos.




(c) **Resultados de FPR.** El clasificador *CollectiveForest* alcanza los mejores resultados, comparado con el rendimiento del resto de clasificadores: un valor de 0,01 etiquetando el 10 % del conjunto de datos.




(d) **Resultados de AUC.** El mejor clasificador es *CollectiveForest*, ya que devuelve un valor de AUC de 0,68 con el 66 % de esfuerzo realizado en la tarea de etiquetado.

**Figura 5.4:** Resultados realizados con las características del modelo VSM empleando *n-gramas* y aplicando la técnica de SMOTE. En consecuencia, el clasificador *CollectiveForest* es el clasificador más completo del conjunto de clasificadores, debido a que ofrece los resultados más elevados tanto en precisión como en FPR y AUC.

Respecto a los resultados obtenidos sin aplicar la técnica de SMOTE, en ambos enfoques VSM y aplicando algoritmos de aprendizaje automático supervisado, el clasificador *Random Forest* con 100 árboles, empleando como modelo VSM los *n-gramas*, es el mejor clasificador. Obtiene resultados similares en precisión (77,20 % contra 79,41 %), TPR (0,16 contra 0,22) y FPR (0,03 contra 0,02) que el clasificador SVM con el núcleo polinomial normalizado, pero en términos de AUC alcanza el mayor valor: 0,70.

En la misma línea de no aplicar SMOTE, el mejor clasificador para el aprendizaje colectivo es *CollectiveForest*, ya que empleando el enfoque VSM basado en *n-gramas* y solamente con el 50 % de las instancias conocidas de todo el conjunto de datos de comentarios, logra un valor de 77,11 % en precisión, de 0,69 en AUC, de 0,13 en TPR y de 0,02 en FPR. Si se comparan los resultados, la clasificación colectiva consigue resultados similares pero reduce considerablemente el esfuerzo de etiquetado a la mitad (el 50 %).

Por otra parte, respecto a los algoritmos supervisados aplicando la técnica de SMOTE, *Random Forest* con un valor de $N = 100$ y con un enfoque VSM basado en palabras, alcanza resultados significativos: un valor de 77,08 % en precisión, un valor de 0,18 en TPR, un valor de 0,04 en FPR y un valor de 0,67 en AUC.

En la clasificación colectiva y aplicando la técnica de SMOTE, el clasificador *CollectiveForest* para un enfoque VSM utilizando las palabras como términos y únicamente etiquetando el 75 % del total de nuestro conjunto de datos, obtiene un valor de precisión de 76,94 %, 0,16 de TPR, 0,04 de FPR y 0,67 de AUC. Los resultados para la clasificación colectiva se aproximan de manera significativa a los resultados obtenidos mediante los enfoques supervisados, mientras que el esfuerzo de etiquetar todo el conjunto de datos de los comentarios se ha reducido en un 25 %.

En conclusión, deducimos que la aplicación de la técnica SMOTE no hace mejorar los resultados obtenidos, aunque dicha técnica balancea las clases y por ello, hace que la fase de entrenamiento de cada algoritmo sea más adecuada y confortable. Con respecto a la utilización de la clasificación colectiva, se observa como es capaz de mantener los resultados del mejor clasificador del aprendizaje automático supervisado, mientras los esfuerzos de etiquetar todo el conjunto de datos de los comentarios se reduce significativamente.

### 5.4.2 Conjunto de usuarios

La Tabla 5.9 muestra los resultados en el enfoque VSM con las palabras, el aprendizaje supervisado y sin aplicar SMOTE, la Tabla 5.10 expone los resultados con *n-gramas*, el aprendizaje supervisado y sin aplicar SMOTE. La Tabla 5.11 muestra los resultados del modelo VSM basado en palabras, el aprendizaje supervisado y la aplicación de SMOTE. La Tabla 5.12 expone los resultados del modelo VSM basado en *n-gramas*, el aprendizaje supervisado y la técnica de SMOTE.

Estas 4 tablas contienen los resultados del aprendizaje supervisado sobre el enfoque VSM basado en palabras, con y sin SMOTE y el enfoque VSM basado en *n-gramas*, con y sin SMOTE.

**Tabla 5.9:** Resultados en términos de precisión, tasa de verdaderos positivos (TPR), tasa de falsos positivos (FPR) y área bajo la curva ROC (AUC) sobre el nivel de controversia del usuario, para VSM empleando palabras.

| Clasificador | Precisión (%) | TPR | FPR | AUC |
|---|---|---|---|---|
| KNN K = 10 | 89,43 ± 4,93 | 0,80 ± 0,12 | 0,04 ± 0,01 | 0,97 ± 0,02 |
| BN: Bayes K2 | 82,52 ± 2,00 | 0,82 ± 0,03 | 0,17 ± 0,02 | 0,87 ± 0,02 |
| BN: Bayes TAN | 90,47 ± 1,47 | 0,99 ± 0,02 | 0,15 ± 0,02 | 0,96 ± 0,01 |
| Naïve Bayes | 67,10 ± 4,33 | 0,28 ± 0,12 | 0,06 ± 0,02 | 0,82 ± 0,03 |
| SVM: Polinomial | 75,50 ± 4,46 | 0,43 ± 0,12 | 0,03 ± 0,01 | 0,70 ± 0,06 |
| SVM: Polinomial normal. | 95,00 ± 1,81 | 0,93 ± 0,04 | 0,04 ± 0,01 | 0,95 ± 0,02 |
| SVM: PVK | 95,47 ± 1,31 | 0,96 ± 0,03 | 0,05 ± 0,02 | 0,96 ± 0,01 |
| SVM: RBF | 59,57 ± 0,31 | 0,00 ± 0,01 | 0,00 ± 0,00 | 0,50 ± 0,00 |
| DT: J48 | 96,58 ± 0,86 | 0,97 ± 0,02 | 0,04 ± 0,01 | 0,97 ± 0,01 |
| DT: Random Forest N=100 | 96,43 ± 0,94 | 0,97 ± 0,02 | 0,04 ± 0,01 | 0,99 ± 0,00 |

**Tabla 5.10:** Resultados en términos de precisión, tasa de verdaderos positivos (TPR), tasa de falsos positivos (FPR) y área bajo la curva ROC (AUC) sobre el nivel de controversia del usuario, para VSM empleando *n-gramas*.

| Clasificador | Precisión (%) | TPR | FPR | AUC |
|---|---|---|---|---|
| KNN K = 10 | 89,55 ± 4,81 | 0,80 ± 0,12 | 0,04 ± 0,01 | 0,97 ± 0,02 |
| BN: Bayes K2 | 82,52 ± 2,00 | 0,82 ± 0,03 | 0,17 ± 0,02 | 0,87 ± 0,02 |
| BN: Bayes TAN | 90,47 ± 1,47 | 0,99 ± 0,02 | 0,15 ± 0,02 | 0,96 ± 0,01 |
| Naïve Bayes | 67,31 ± 4,63 | 0,27 ± 0,12 | 0,05 ± 0,02 | 0,82 ± 0,03 |
| SVM: Polinomial | 75,56 ± 4,43 | 0,44 ± 0,12 | 0,03 ± 0,01 | 0,70 ± 0,06 |
| SVM: Polinomial normal. | 95,09 ± 1,59 | 0,94 ± 0,04 | 0,04 ± 0,01 | 0,95 ± 0,02 |
| SVM: PVK | 95,68 ± 1,13 | 0,96 ± 0,03 | 0,05 ± 0,02 | 0,96 ± 0,01 |
| SVM: RBF | 59,57 ± 0,31 | 0,00 ± 0,01 | 0,00 ± 0,00 | 0,50 ± 0,00 |
| DT: J48 | 96,58 ± 0,86 | 0,97 ± 0,02 | 0,04 ± 0,01 | 0,97 ± 0,01 |
| DT: Random Forest N=100 | 96,49 ± 0,97 | 0,97 ± 0,02 | 0,04 ± 0,01 | 0,99 ± 0,00 |

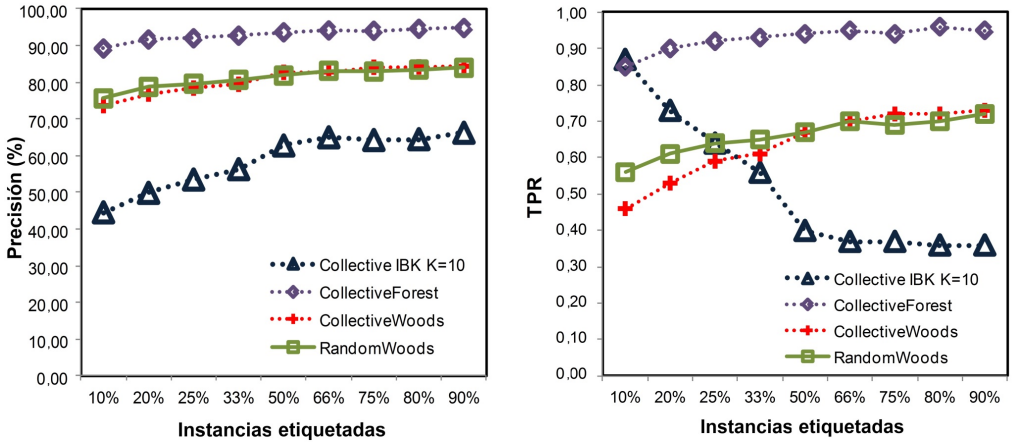
**Tabla 5.11:** Resultados en términos de precisión, TPR, FPR y AUC para el nivel de controversia de los usuarios, para el enfoque VSM basado en palabras, aplicando SMOTE.

| Clasificador | Precisión (%) | TPR | FPR | AUC |
|---|---|---|---|---|
| KNN K = 10 | 90,92 ± 3,65 | 0,87 ± 0,08 | 0,06 ± 0,02 | 0,97 ± 0,02 |
| BN: Bayes K2 | 73,71 ± 3,05 | 0,58 ± 0,07 | 0,15 ± 0,02 | 0,86 ± 0,02 |
| BN: Bayes TAN | 89,67 ± 2,10 | 0,91 ± 0,03 | 0,11 ± 0,03 | 0,97 ± 0,01 |
| Naïve Bayes | 68,41 ± 7,91 | 0,44 ± 0,21 | 0,15 ± 0,24 | 0,88 ± 0,02 |
| SVM: Polinomial | 86,84 ± 2,22 | 0,75 ± 0,05 | 0,05 ± 0,01 | 0,85 ± 0,03 |
| SVM: Polinomial normal. | 95,42 ± 0,89 | 0,96 ± 0,02 | 0,05 ± 0,01 | 0,96 ± 0,01 |
| SVM: PVK | 95,30 ± 1,00 | 0,98 ± 0,02 | 0,07 ± 0,02 | 0,96 ± 0,01 |
| SVM: RBF | 61,06 ± 1,19 | 0,11 ± 0,10 | 0,04 ± 0,05 | 0,53 ± 0,02 |
| DT: J48 | 96,55 ± 0,87 | 0,97 ± 0,02 | 0,04 ± 0,01 | 0,97 ± 0,01 |
| DT: Random Forest N=100 | 96,46 ± 0,99 | 0,97 ± 0,02 | 0,04 ± 0,01 | 0,99 ± 0,00 |

**Tabla 5.12:** Resultados en términos de precisión, TPR, FPR y AUC para el nivel de controversia de los usuarios, para el enfoque VSM basado en *n-gramas*, aplicando SMOTE.

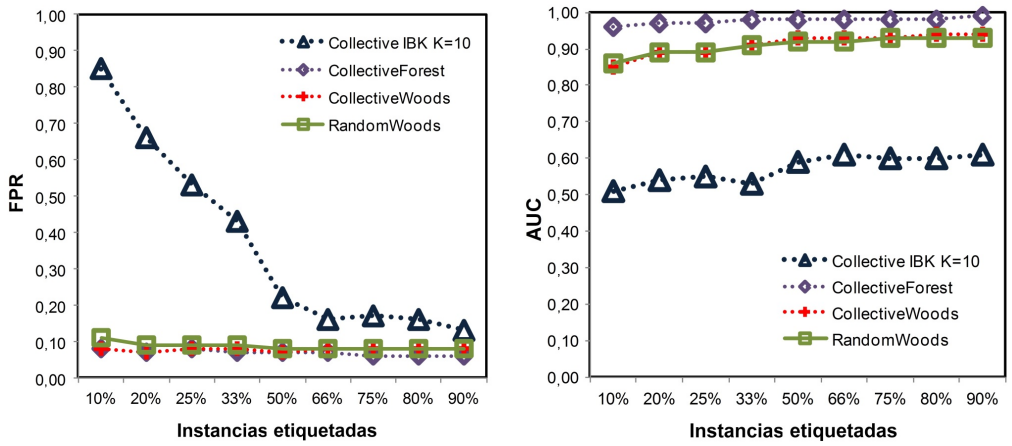
| Clasificador | Precisión (%) | TPR | FPR | AUC |
|---|---|---|---|---|
| KNN K = 10 | 90,86 ± 3,78 | 0,86 ± 0,09 | 0,06 ± 0,02 | 0,97 ± 0,02 |
| BN: Bayes K2 | 73,95 ± 2,83 | 0,58 ± 0,06 | 0,15 ± 0,02 | 0,87 ± 0,02 |
| BN: Bayes TAN | 89,82 ± 2,32 | 0,90 ± 0,03 | 0,11 ± 0,03 | 0,97 ± 0,01 |
| Naïve Bayes | 68,56 ± 4,19 | 0,30 ± 0,11 | 0,05 ± 0,02 | 0,88 ± 0,02 |
| SVM: Polinomial | 86,81 ± 2,14 | 0,74 ± 0,05 | 0,05 ± 0,01 | 0,85 ± 0,03 |
| SVM: Polinomial normal. | 95,50 ± 0,82 | 0,96 ± 0,02 | 0,05 ± 0,01 | 0,96 ± 0,01 |
| SVM: PVK | 95,56 ± 1,11 | 0,99 ± 0,01 | 0,06 ± 0,02 | 0,96 ± 0,01 |
| SVM: RBF | 61,36 ± 1,47 | 0,10 ± 0,09 | 0,04 ± 0,04 | 0,53 ± 0,03 |
| DT: J48 | 96,70 ± 0,92 | 0,97 ± 0,02 | 0,04 ± 0,01 | 0,98 ± 0,01 |
| DT: Random Forest N=100 | 96,28 ± 1,01 | 0,97 ± 0,02 | 0,04 ± 0,01 | 0,99 ± 0,00 |

La Figura 5.5 muestra los resultados con el enfoque VSM generado mediante palabras como términos, haciendo uso del aprendizaje colectivo y sin aplicar la técnica de SMOTE; la Figura 5.6 expone los resultados con el modelo VSM implementado mediante *n-gramas* como términos del modelo, utilizando el aprendizaje semi-supervisado colectivo y sin el método SMOTE aplicado; la Figura 5.7 muestra los resultados con el modelo VSM generado mediante palabras, empleando los algoritmos que el aprendizaje colectivo nos aporta y aplicando el método de SMOTE y por último, la Figura 5.8 expone los resultados que el enfoque VSM generado con *n-gramas* como términos en el modelo, el uso del aprendizaje semi-supervisado colectivo y aplicando la técnica de SMOTE nos aporta.




(a) **Resultados de precisión.** El clasificador *CollectiveForest*, con un 90 % de etiquetado del conjunto de datos de usuarios, logra alcanzar un valor de 94,76 %, siendo el mejor clasificador para esta medida.

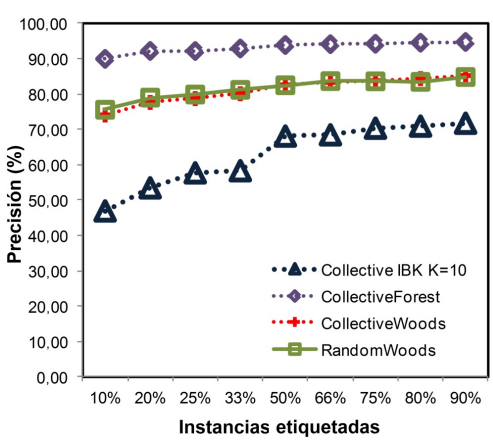
(b) **Resultados de TPR.** El mejor clasificador es *CollectiveForest*, con una precisión de 96 % en términos de TPR mediante el etiquetado del 80 % del total de conjunto de datos.



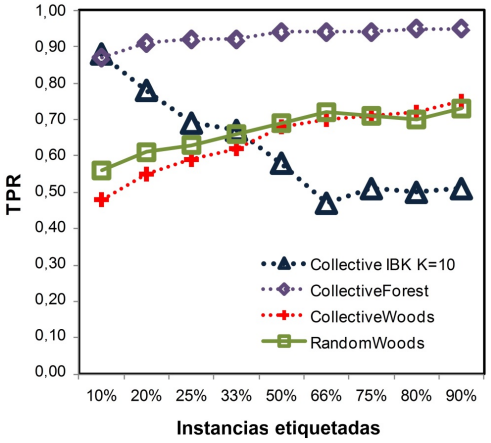

(c) **Resultados de FPR.** La mayoría de los clasificadores, excepto *Collective IBK*, devuelven resultados notables, todos ellos alrededor de 0. El clasificador *CollectiveWoods* obtiene valores cercanos a 0 y únicamente con el 20 % del esfuerzo de etiquetado realizado.

(d) **Resultados de AUC.** Los resultados de AUC obtenidos son muy similares entre ellos. El clasificador *CollectiveForest* ofrece una salida de 0,98 solamente etiquetando el 33 % de las instancias del conjunto de datos.

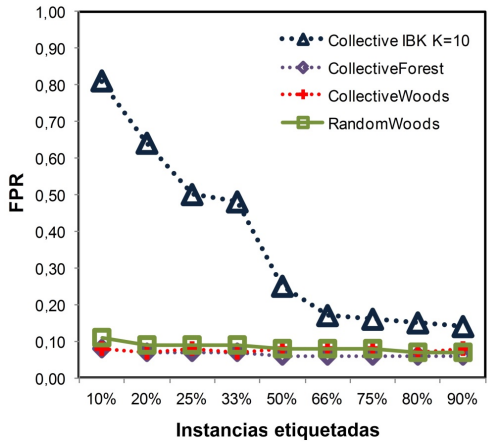
**Figura 5.5:** Resultados para nuestra clasificación colectiva para la categorización de usuarios usando palabras como términos en el enfoque VSM. En resumen, el clasificador *CollectiveForest* es el mejor clasificador en este ámbito.



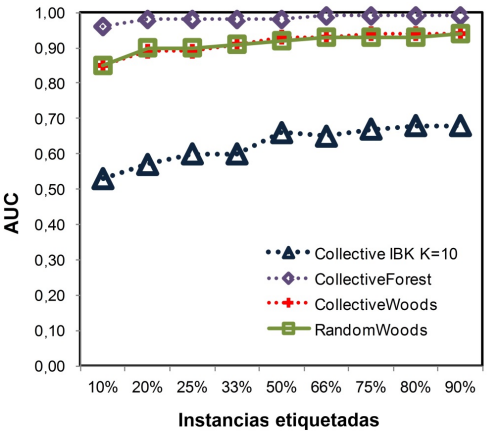

(a) **Resultados de precisión.** El clasificador *CollectiveForest*, utilizando un 90 % de las instancias etiquetadas del conjunto de datos, consigue un valor de 94,70 % de precisión.




(b) **Resultados de TPR.** El clasificador *CollectiveForest* es el mejor clasificador para TPR obteniendo un valor de 0,85 etiquetando el 80 % del total de conjunto de datos de usuarios.

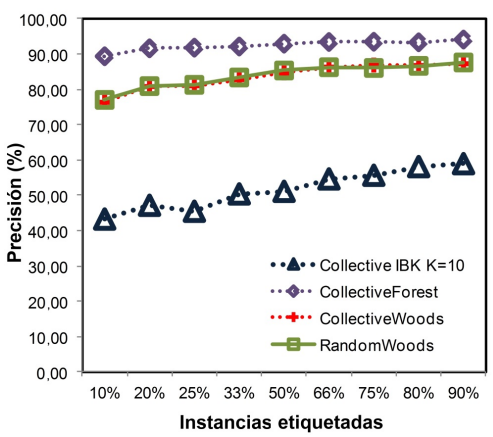



(c) **Resultados de FPR.** El clasificador *CollectiveForest* consigue lograr resultados muy cercanos al valor 0, así como el resto de clasificadores excepto *Collective IBK*.

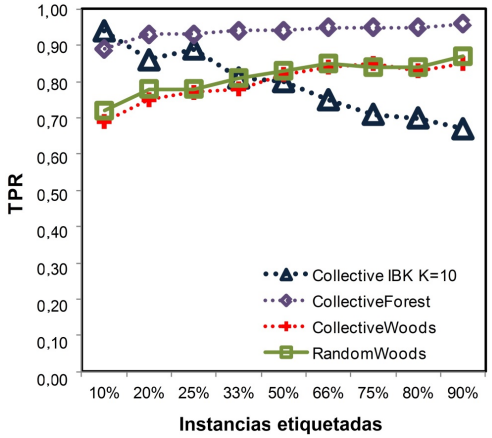



(d) **Resultados de AUC.** Respecto a los resultados de AUC obtenidos, el mejor algoritmo es *CollectiveForest*. Este clasificador alcanza un valor de 0,99 cuando únicamente se ha etiquetado el 66 % del total del conjunto de usuarios descargado.

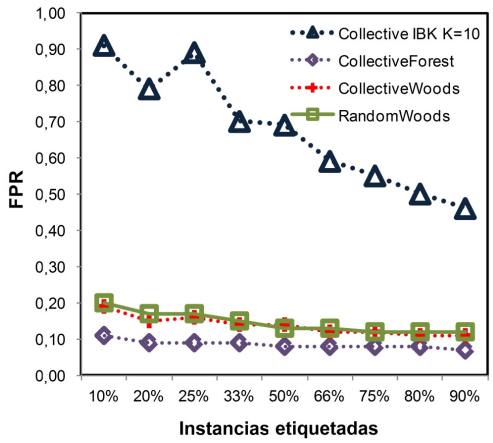
**Figura 5.6:** Resultados sobre nuestra clasificación colectiva para la categorización de usuarios empleando el modelo de VSM basado en *n-gramas* como términos del modelo. Resumiendo, el algoritmo *CollectiveForest* es el clasificador que mejores resultados obtiene para dicha clasificación.



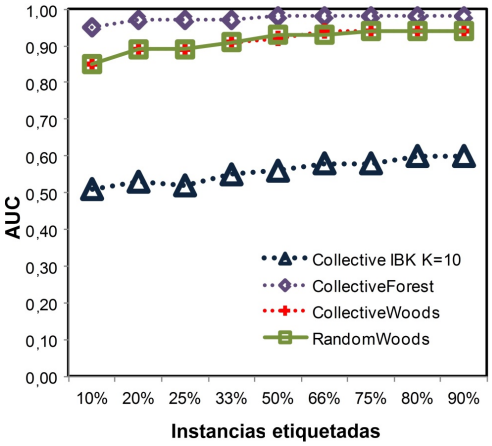

(a) **Resultados de precisión.** El algoritmo *CollectiveForest* obtiene el mayor valor de precisión: 94,08 % etiquetando el 90 % de las instancias del conjunto de datos de usuarios.




(b) **Resultados de TPR.** El clasificador *Collective IBK* logra resultados notables para un número mínimo de instancias etiquetadas, con un valor de 0,94 empleando solamente el 10 % del conjunto de datos.

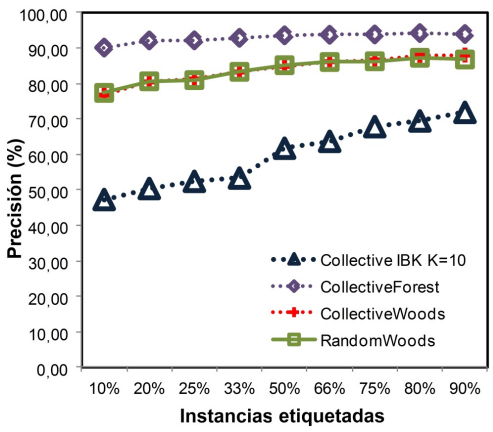



(c) **Resultados de FPR.** Todos los clasificadores, excepto *Collective IBK*, ofrecen resultados considerables. El clasificador *CollectiveForest* obtiene el mejor resultado con un valor de 0,08 etiquetando únicamente la mitad de las instancias.

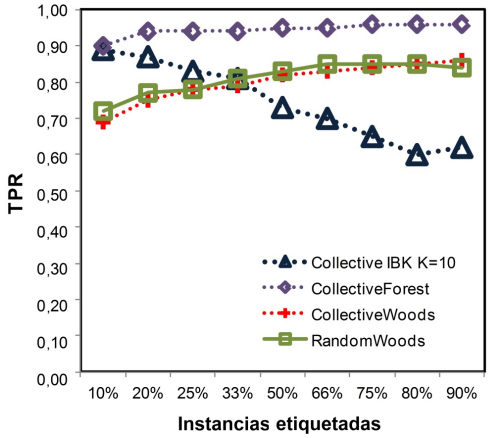



(d) **Resultados de AUC.** El clasificador *CollectiveForest*, etiquetando un 50 % de las instancias sobre el total del conjunto de comentarios, logra alcanzar un valor de 0,98.

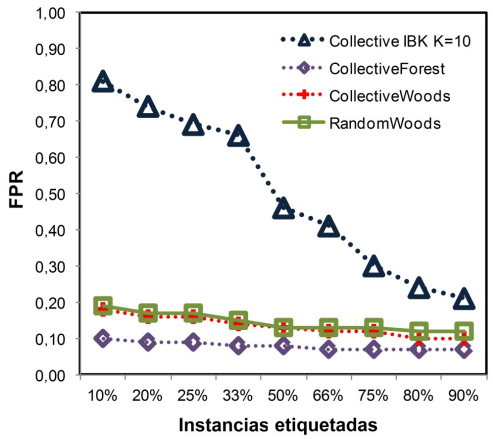
**Figura 5.7:** Resultados realizados con las características del enfoque VSM empleando palabras y aplicando la técnica de SMOTE para categorización de usuarios. Consecuentemente, el clasificador *CollectiveForest* logró los resultados más destacados en diferentes medidas como la precisión, el FPR y la AUC.



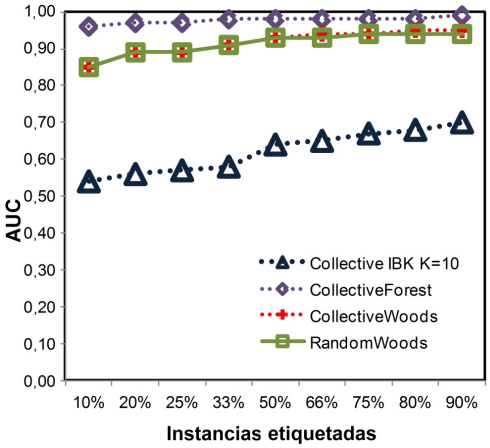

(a) **Resultados de precisión.** El mejor resultado es el logrado por el clasificador *CollectiveForest* etiquetando el 80 % del conjunto de datos: un valor de 94,20 % de precisión.




(b) **Resultados de TPR.** El clasificador *Collective IBK* logra obtener el mejor resultado de entre todos los demás clasificadores: un valor de 0,89, etiquetando exclusivamente el 10 % del total del conjunto de datos.




(c) **Resultados de FPR.** El clasificador *CollectiveForest* alcanza los mejores resultados comparado con el rendimiento del resto de clasificadores, un valor de 0,07 etiquetando el 66 % del conjunto de datos.




(d) **Resultados de AUC.** El mejor clasificador es *CollectiveForest* ya que devuelve un valor de AUC de 0,98 con el 33 % de esfuerzo realizado en la tarea de etiquetar.

**Figura 5.8:** Resultados realizados con las características del modelo VSM empleando *n-gramas* y aplicando la técnica de SMOTE para la categorización de usuarios. En consecuencia, el clasificador *CollectiveForest* es el clasificador más completo del conjunto de clasificadores, debido a que ofrece los resultados más elevados tanto en precisión como en FPR y AUC.

En cuanto a los resultados obtenidos tanto con el enfoque VSM basado en palabras como el basado en *n-gramas*, aplicando los algoritmos supervisados y las clasificaciones colectivas, el clasificador *Random Forest* con una implementación de 100 árboles empleando *n-gramas* como enfoque VSM es el mejor clasificador. Obtiene resultados similares en precisión (96,49 % contra 96,70 %), en TPR (0,97 contra 0,97) y en FPR (0,04 contra 0,04) que el clasificador J48, pero en términos de AUC alcanza el valor más elevado: 0,99.

En este sentido, el clasificador *CollectiveForest* empleando el enfoque VSM basado en *n-gramas* unicamente con el 66 % de las instancias conocidas del total del conjunto de datos, obtiene un valor de precisión de 94,08 %, un valor de AUC de 0,99, un valor de TPR de 0,94 y un valor de FPR de 0,06. Con respecto a la utilización de la clasificación colectiva, si comparamos los resultados alcanzados por los enfoques supervisados, esta clasificación colectiva obtiene resultados cercanos. Por lo tanto, demostramos que podemos mantener los mejores resultados del mejor algoritmo del aprendizaje automático supervisado mientras que los esfuerzos del etiquetado son reducidos de manera significativa, en este caso en un tercio, es decir etiquetando únicamente el 66 % del total del conjunto de datos.

Por otro lado, haciendo referencia a los algoritmos supervisados y aplicando la técnica de SMOTE, el clasificador J48, perteneciente a la familia de los árboles de decisión, mediante el empleo de *n-gramas* en el enfoque VSM, logra unos resultados óptimos con un valor de precisión del 96,70 %, un valor de TPR de 0,97, un valor de FPR de 0,04 y un valor de AUC de 0,98.

Si hacemos mención de la clasificación colectiva al aplicar la técnica de SMOTE, el clasificador *CollectiveForest* gracias a un enfoque VSM basado en *n-gramas* y solamente etiquetando el 33 % de las instancias del conjunto de datos perteneciente a los usuarios, logra alcanzar unos valores de precisión de 92,86 %, un valor de 0,94 de TPR, un valor de 0,08 de FPR y un valor de 0,98 de AUC. Estos resultados logrados con el aprendizaje colectivo son bastante similares a los obtenidos con el aprendizaje automático supervisado, pero con la salvedad de que gracias a este tipo de aprendizaje el esfuerzo a la hora de etiquetar nuestro conjunto de usuarios se ha reducido en un 66 %, es decir, únicamente necesitamos etiquetar el 33 % del conjunto de datos.

La conclusión obtenida es que a la hora de aplicar la técnica de SMOTE, la cual equilibra ambas clases en nuestro conjunto de usuarios, hemos logrado resultados bastante parejos respecto a las clasificaciones realizadas sin aplicar dicha técnica. Por ello, deducimos que no es necesario proceder a aplicar SMOTE, ya que aumenta considerablemente el tiempo de computación de los expe-

rimentos al tener que balancear las clases que componen nuestro conjunto de datos, obteniendo resultados similares sin su aplicación. Respecto al empleo del aprendizaje semi-supervisado y su clasificación colectiva, es notorio cómo esta clasificación puede igualar los mejores resultados del conjunto de clasificadores supervisados, a la vez que se reduce considerablemente las instancias etiquetadas necesarias dentro del conjunto de datos de los usuarios y por lo tanto, el esfuerzo de realizar dicho aprendizaje y etiquetado.

## 5.5 Sumario

El capítulo 5 desarrolla la clasificación colectiva, perteneciente al aprendizaje semi-supervisado, como mejora de la categorización implementada con el aprendizaje supervisado en el capítulo 4; ya que a la hora de comparar ambas clasificaciones hemos conseguido mantener los resultados del aprendizaje supervisado, pero reduciendo el esfuerzo de etiquetado notablemente.

Primeramente, hemos realizado una introducción, con el fin de entrar en materia respecto al concepto de clasificación colectiva, a la vez que hemos expuesto las aportaciones y contribuciones implementadas en este capítulo.

Seguidamente, hemos descrito el método empleado a la hora de categorizar tanto el conjunto de comentarios como el conjunto de usuarios, gracias a las características estadísticas, sintácticas y de opinión extraídas de ambos conjuntos de datos.

En tercer lugar, hemos explicado los diferentes tipos de clasificadores y algoritmos colectivos empleados en esta investigación, con el fin de categorizar ambos conjunto de datos: (i) el conjunto de comentarios y (ii) el conjunto de usuarios con sus respectivos comentarios.

A continuación, hemos expuesto la validación empírica realizada, a fin de mejorar los resultados obtenidos en las clasificaciones ejecutadas anteriormente con los clasificadores y algoritmos pertenecientes al aprendizaje supervisado. Para ello, hemos seguido una metodología concreta, en la cual hemos aplicado una serie de técnicas y enfoques: (i) la validación cruzada, (ii) el *Information Gain* y (iii) el SMOTE, así como los diferentes algoritmos semi-supervisados. Para evaluar los resultados, hemos tenido distintas métricas como: (i) el TPR, (ii) el FPR, (iii) la precisión y (iv) la AUC. Al igual que en capítulo 4, en primer lugar hemos procedido a clasificar los comentarios, para posteriormente y en base a estos resultados, categorizar los usuarios con sus respectivos comentarios siguiendo la misma metodología.

En último lugar, hemos mostrado los resultados logrados por la clasificación colectiva mediante diferentes representaciones gráficas según la aplicación de la técnica SMOTE y según el enfoque empleado de VSM, tanto el basado en palabras como en *n-gramas*, para poder compararlos con los resultados obtenidos mediante el aprendizaje supervisado (mostrados en diferentes tablas). Una vez expuestos tales resultados, hemos presentado los mejores valores ofrecidos por dichos algoritmos, tanto supervisados como semi-supervisados, así como las conclusiones extraídas acerca de la utilización de la clasificación colectiva en nuestra investigación para nuestros conjuntos de datos.

*«Sólo podemos ver un poco del futuro, pero lo suficiente para darnos cuenta de que hay mucho que hacer.»*

Alan Mathison Turing (1912–1954)

CAPÍTULO

# 6

# Análisis y predicción de series temporales

La historia referente a los métodos de información evoluciona en base a los desarrollos científicos estadísticos, concretamente, al análisis de procesos estocásticos y series temporales. Los antecedentes científicos de la predicción de series temporales se pueden hallar a mediados del siglo XIX, teniendo como temática principal los estudios meteorológicos y demográficos. El autor Verhulst (1845) presentó distintos trabajos sobre la variación de la densidad poblacional en forma de curva logística. Por otra parte, Buys-Ballot (1847) realizó experimentos sobre los cambios periódicos de temperatura, mientras que Beveridge (1922) elaboró un gráfico sobre la evolución periódica del trigo en Europa.

Con el transcurso de los años, se empiezan a investigar otros temas, como la economía. En este caso y debido a las fluctuaciones que se pueden observar en las series económicas, se producen una serie de desarrollos en esta materia de gran trascendencia. Yule (1927) y Slutzky (1937) desarrollaron las primeras investigaciones sobre trabajos autorregresivos y medias móviles.

Una vez acabada la Segunda Guerra Mundial, se empiezan a acelerar las implementaciones de técnicas de predicción. Los investigadores Burns y Mitchell (1946) publican un artículo sobre los ciclos económicos y la utilización de indicadores. A su vez, Shiskin y Eisenpress (1957) presentan un trabajo sobre la

descomposición de series (componentes cíclicos, estacionales y tendenciales). Por su parte, Kalman *et al*. (1960) publican una investigación sobre el filtrado de series, y gracias a los artículos de Tukey (1961) y Kolmogorov (1941) nace el análisis espectral. Siguiendo esta línea, gracias a las contribuciones de Holt (1957) y Winters (1960), aparecen las técnicas de alisado exponencial y como resultado de la publicación conjunta de Box y Jenkins (1970), los modelos ARIMA.

En un ámbito diferente como es el campo de la psicología, una técnica muy conocida es el método *Delphi*. Desarrollado para aplicaciones militares por la *Rand Corporation* y publicado a *posteriori* por sus autores Dalkey y Helmer (1963). En el mismo campo, tanto los desarrollos de *Herbert Simon* (Premio Nobel de Economía de 1978) como diversos trabajos de psicología aplicada fijaron las bases para obtener la información mediante otros procedimientos específicos. El matemático Neyman (1938), mediante desarrollos estadísticos en muestras de poblaciones finitas, inicia el auge de múltiples encuestas de opinión, actitudes o expectativas, que se implementarán sistemáticamente a partir del autor Tobin (1959).

Una vez hecha una breve introducción histórica sobre la predicción y las series temporales, se muestra una definición formal sobre las mismas: una serie temporal es una secuencia de observaciones sobre una variable aleatoria y por lo tanto, es un proceso estocástico. La predicción de datos mediante series temporales es un componente importante en la investigación de operaciones, ya que estos datos, a menudo, son la base de los modelos de decisión.

El análisis de series temporales proporciona las herramientas necesarias para la selección de un modelo, que pueda ser usado para predecir los futuros eventos. Esta selección debe ser cuidadosa, ya que el modelado de las series temporales es un problema estadístico. Las predicciones se utilizan en métodos computacionales para estimar los parámetros de un modelo, el cual se emplea con recursos asignados limitados o para describir procesos aleatorios.

El único modo de obtener predicciones no es solamente el análisis de series temporales. Comúnmente, se utiliza el juicio de expertos para predecir cambios a largo plazo en la estructura de un sistema. Por ejemplo, los métodos cualitativos (como la técnica *Delphi*) se pueden usar para predecir las grandes innovaciones tecnológicas y sus efectos.

El propósito principal de modelar una serie temporal es hacer predicciones, las cuales más adelante puedan ser utilizadas directamente en la toma de decisiones, como ordenar las reposiciones de un sistema de inventariado o desarrollar los horarios del personal para el correcto funcionamiento de una planta de

producción. También pueden ser utilizadas como parte de un modelo matemático para un análisis de decisiones más complejo.

Con estos antecedentes, se pretende modelar el comportamiento de los usuarios en los sitios web sociales mediante el empleo de técnicas de predicción y series temporales, con el fin de pronosticar las diferentes acciones que el usuario va a tomar en el propio sitio web social y así prevenir comportamientos ilícitos o agravantes. Es decir, poder adelantarse a las acciones que vayan a ser realizadas por los usuarios tipo trol.

Resumiendo, nuestras principales aportaciones son las siguientes:

- Un nuevo método para modelar el comportamiento de usuarios y sus comentarios en sitios web sociales.
- Una adaptación de las técnicas de predicción y series temporales, a fin de filtrar comentarios y usuarios trol, gracias a las características extraídas de sus respectivos perfiles públicos de *Menéame*.
- Una validación empírica, la cual demuestra que nuestro método puede lograr bajas tasas de error en la predicción del comportamiento de usuarios.

El resto del capítulo está estructurado de la siguiente manera. La sección 6.1 describe el método empleado para extraer las características tanto del conjunto de datos de comentarios como del conjunto de datos de usuarios procedentes de *Menéame* y así elaborar un nuevo conjunto de datos, con el fin de poder aplicar técnicas de predicción y series temporales. La sección 6.2 hace referencia a los diferentes modelos predictivos de series temporales empleados en esta tesis. La sección 6.3 desarrolla la validación empírica para nuestra metodología, en cuanto a la predicción del comportamiento de los usuarios en el sitio web social *Menéame* y en base a ciertas características extraídas de los perfiles públicos. La sección 6.4 hace referencia a los resultados y conclusiones obtenidas en este capítulo sobre el modelado y la predicción de usuarios en sitios web sociales. Por último, la sección 6.5 resume los conceptos expresados a lo largo de este capítulo.

## 6.1 Descripción del método

Al igual que en el aprendizaje supervisado y semi-supervisado empleados con anterioridad para categorizar a los usuarios y sus comentarios, en las técnicas de

predicción y series temporales ha sido necesario etiquetar y clasificar en clases los conjuntos de datos de los usuarios.

Para ello, hemos procedido a descargar todos los comentarios (y sus respectivas noticias) de 13 usuarios elegidos aleatoriamente en base al número de comentarios y al tipo de usuario al que pertenecen, desde el día en el que publicaron el primer comentario hasta el 25 de Septiembre de 2012. En la Tabla 6.1, se muestran los 13 usuarios seleccionados con sus características:

**Tabla 6.1:** Lista de usuarios descargados con el fin de aplicar series temporales.

| Nombre de usuario | Número de comentarios | Tipo de usuario |
|---|---|---|
| gabi10 | 206 | Lector |
| pavlenka | 217 | Lector |
| batis | 375 | Lector |
| hangla | 349 | Lector |
| willies | 203 | Comentarista |
| tachyon | 359 | Comentarista |
| berthax | 224 | Comentarista |
| hsc | 315 | Comentarista |
| psalbendea | 306 | Comentarista |
| AmadorCea | 108 | Participativo |
| anadelagua | 517 | Participativo |
| alfonsoblog | 129 | Participativo |
| arolasecas | 147 | Participativo |

A continuación, la Figura 6.1 muestra un extracto del archivo de un usuario formado por el título, el resumen, las etiquetas, las categorías y los comentarios que el usuario publicó en la noticia.

### 6.1.1 **Características extraídas**

Esta sección describe cuáles son las características que van a ser empleadas en la generación del conjunto de datos, con el fin de aplicar métodos de series temporales y así predecir el comportamiento de los usuarios. Hemos dividido estas características en 5 categorías diferentes: (i) las características estadísticas, (ii) las características sintácticas, (iii) las características de opinión, (iv) las características referentes al comentario y (v) las características referentes al usuario (explicadas más detalladamente en la sección 3.3.1 y en la sección 3.3.2):

TITULO: Un joven presencia el asesinato de su novia a través de una webcam
RESUMEN: Una joven china ha sido asesinada en la Universidad de York en Toronto mientras hablaba con su novio a través de una cámara web. Su pareja, residente en China, ha presenciado como un desconocido entraba en la habitación de la víctima y forcejeaba con ella. Relacionada con esta noticia en inglés: www.meneame.net/story/estudiante-china-muerta-canada-despues-novio-vie
etiquetas: joven, asesinato, novia, webcam
CATEGORIAS: actualidad, sucesos

COMENTARIO: #8 Yo ya había dicho que era igual que la otra noticia pero en español, tampoco es para que os ensañeis a votar negativamente como duplicada y encima cuando ya no se puede quitar la noticia 6 votos: 0 el 20-04-2011 13:40 UTC por alfonsoblog Num Respuestas: 0

COMENTARIO: #6 #5 Si bueno, lo único es que aquí está la información en castellano, queria subirla para quien le fuera más cómodo leerla así 6 votos: 0 el 20-04-2011 13:10 UTC por alfonsoblog Num Respuestas: 0

FIN-NOTICIA

**Figura 6.1:** Extracto del archivo de un usuario.

- **Características estadísticas.**
  - *Cuerpo del comentario.* Texto escrito por el usuario donde expresa su opinión sobre la noticia o sobre un comentario. De longitud variable.
  - *Título.* Es el título de la noticia (el mismo que el de la historia externa al que hace referencia).
  - *Resumen.* Es una pequeña descripción de la noticia, la cual es identificativa y aclarativa con la noticia.

  - *Etiquetas.* Son las palabras clave de la noticia relacionadas con el contenido del enlace, no con la temática de la misma. El número aproximado de etiquetas es de 4 o 5 palabras y las define el usuario.
  - *Categorías.* Temáticas gracias a las cuales se agrupan las noticias (ciencia, política, actualidad, deportes, etc.). Son definidas por el usuario.
  - *Fecha.* Es el valor que indica hace cuanto tiempo el usuario de *Menéame* publicó el comentario.
  - *Número de referencias al comentario.* Indica el número de veces que el comentario ha sido referenciado en otros comentarios de la misma noticia.
  - *Número de referencias desde el comentario.* Mide el número de referencias del comentario a otros comentarios dentro de la misma noticia.
  - *Número de comentario.* Indica la antigüedad del comentario en la noticia.
  - *Similitud del comentario con el resumen de la noticia.* Hemos empleado la similitud del enfoque VSM del comentario con el modelo del resumen de la noticia. En particular, hemos usado la similitud del coseno (Tata y Patel, 2007).
  - *Número de coincidencias entre las palabras del comentario y las etiquetas de la noticia.* Hemos contado el número de palabras que aparecen en el comentario y que aparecen, a su vez, como etiquetas de la historia.
  - *Número de enlaces en el cuerpo del comentario.* Hemos contabilizado el número de enlaces dentro del cuerpo del comentario.

- **Características sintácticas.** Hemos realizado un etiquetado de POS (acrónimo del inglés *Part of Speech*) haciendo uso de la herramienta *Freeling*[1]. Han sido utilizadas las siguientes características:

  - Número de adjetivos en el cuerpo del comentario.
  - Número de números en el cuerpo del comentario.
  - Número de fechas en el cuerpo del comentario.
  - Número de adverbios en el cuerpo del comentario.

[1]Disponible en `http://www.lsi.upc.edu/~nlp/freeling`

  - Número de nombres en el cuerpo del comentario.
  - Número de conjunciones en el cuerpo del comentario.
  - Número de pronombres en el cuerpo del comentario.
  - Número de signos de puntuación en el cuerpo del comentario.
  - Número de interjecciones en el cuerpo del comentario.
  - Número de determinantes en el cuerpo del comentario.
  - Número de abreviaturas en el cuerpo del comentario.
  - Número de preposiciones en el cuerpo del comentario.
  - Número de verbos en el cuerpo del comentario.

- **Características de opinión.**

  - *Número de palabras positivas y negativas.* Hemos empleado un léxico de opinión externo[1]. Debido a que los términos en este léxico están escritos en Inglés y *Menéame* está desarrollado en castellano, hemos traducido este léxico al castellano.
  - *Número de votos.* El número de votos positivos del comentario.
  - *Karma.* Hemos recuperado el *karma* computado por el sitio web social *Menéame*.

- **Características referentes al comentario.** Gracias a la categorización realizada previamente sobre el conjunto de comentarios, disponemos de información sobre los propios comentarios como el tipo de comentario al que pertenecen o el grado de polemicidad de los mismos.

  - *Valor sobre el tipo de comentario.* Se fijará el valor 0, 1 o 2 dependiendo del tipo de comentario: (i) opinión, (ii) irrelevante o (iii) aporte, respectivamente.
  - *Valor sobre el grado del comentario.* Se asignará el valor 0, 1 o 2 dependiendo del grado de controversia del comentario: (i) normal o gracioso, (ii) polémico o (iii) muy polémico, respectivamente.

- **Características referentes al usuario.** Gracias a la clasificación realizada anteriormente sobre el conjunto de usuarios, disponemos de información sobre el tipo de usuario que realiza los comentarios.

[1] Disponible en: `http://www.cs.uic.edu/~liub/FBS/opinion-lexicon-English.rar`

– *Tipo de usuario.* La clase de usuario estará determinada por las siguientes opciones: (i) lector, (ii) comentarista, (iii) participativo o (iv) contribuyente.

## 6.2 Modelos predictivos de series temporales

La selección de un método de predicción es una tarea dificultosa que debe ser la base del conocimiento relativo a la cantidad que se está prediciendo.

Para cada modelo considerado hay cientos de variaciones propuestas en la literatura, con lo que sería una tarea costosa el considerar todas las variaciones existentes. Por ello, hemos considerado las versiones básicas de cada modelo (sin las añadiduras o las modificaciones propuestas por algunos investigadores). Por ejemplo en el modelo KNN, hemos empleado la forma básica donde las salidas previstas de los $K$-vecinos son igualmente ponderadas. Hay muchas otras variaciones, como la distancia ponderada KNN, la métrica flexible KNN, la distancia basada en métricas no euclidianas y así sucesivamente, pero no hemos considerado estas variantes.

La razón por la que hemos seleccionado estos 5 modelos es que son algunos de los modelos más utilizados en la literatura sobre la predicción de series temporales. A continuación, hemos desarrollado una breve descripción de los modelos considerados.

### 6.2.1 El perceptrón multicapa

El perceptrón multicapa (MLP, de las voces inglesas *Multilayer Perceptron*) es quizás la arquitectura de red en uso más popular hoy en día, tanto para la clasificación como para la regresión (Bishop, 1995). la definición matemática del MLP se indica en la ecuación 6.1.

$$\hat{y} = v_o + \sum_{j=1}^{NH} v_j g(w_j^T x') \tag{6.1}$$

donde los términos de la ecuación se expresan a continuación:

- El término $x'$ es el vector de entrada $x$, aumentado en 1, es decir $x' = (1, x^T)^T$.
- El término $w_j$ es el vector de ponderación del nodo oculto $j$.

- Los términos $v_0$,$v_1$,...,$v_{NH}$ son los pesos para el nodo de salida.
- El término $\hat{y}$ es la salida de la red.
- La función $g$ representa la salida del nodo oculto y es dada en términos de una función sigmoidal[1], por ejemplo la función logística: $g(u) = \frac{1}{1+exp(-u)}$.

El MLP es un modelo muy parametrizado, que gracias a la selección de los nodos ocultos $NH$ podemos controlar la complejidad del modelo. El avance que dio crédito a las redes neuronales sobre su capacidad es la propiedad de aproximación universal (Cybenko, 1989; Funahashi, 1989; Hornik *et al.*, 1989; Leshno *et al.*, 1993). Bajo algunas leves condiciones en las funciones de los nodos ocultos $g$, cualquier función continua dada en un conjunto compacto puede ser aproximada usando una red con un número finito de nodos ocultos.Este es un resultado tranquilizador, ya que es fundamental para evitar la sobre-parametrización, especialmente en las aplicaciones de predicción que por lo general tienen una cantidad limitada de datos muy ruidosos. La selección del modelo (mediante la selección del número de nodos ocultos) ha atraído un gran interés en la literatura de redes neuronales (Medeiros *et al.*, 2006; Anders y Korn, 1999).

Para obtener los pesos se define el error cuadrático medio, mientras que los pesos son optimizados empleando técnicas de gradiente. El método más conocido, basado en el concepto de máxima pendiente[2], es el algoritmo de retropropagación. Un segundo método de optimización denominado *Levenberg-Marquardt*[3] (Levenberg, 1944) es conocido generalmente por ser más eficiente que el algoritmo básico de retropropagación.

### 6.2.2 **Las redes neuronales basadas en funciones de base radial**

Las redes basadas en funciones de base radial (RBF, de las voces inglesas *Radial Basis Function Neural Network*) son similares, en la arquitectura, a la redes

[1] La función sigmoidal permite describir la evolución temporal de muchos procesos naturales y curvas de aprendizaje de sistemas complejos desde unos niveles bajos al inicio, hasta aproximarse a un clímax pasado cierto tiempo; la transición es producida en una región caracterizada por una fuerte aceleración intermedia.

[2] Se denomina recta de máxima pendiente de un plano, a cualquier recta del plano que sea perpendicular a su traza horizontal.

[3] El algoritmo *Levenberg-Marquardt* proporciona una solución numérica al problema de minimizar una función, por lo general no lineal, durante un espacio de parámetros de la función.

multicapa excepto que los nodos tienen una función de activación[1] localizada (Powell, 1987; Moody y Darken, 1989). Por lo general, las funciones de nodo son escogidas como funciones *gaussianas*, con la anchura de la función *gaussiana* controlando la uniformidad de la función ajustada. Las salidas de los nodos son combinadas linealmente para dar la salida final de la red. Específicamente, la salida viene dada por la ecuación 6.2.

$$y = \sum_{j=1}^{NB} w_j e^{-\frac{||x-c_j||^2}{\beta^2}} \tag{6.2}$$

donde $w_j$, $\beta$ y $c_j$ denotan el peso combinado, la anchura de la función de nodo y el centro de la función de nodo de la unidad $j$, respectivamente. Debido a la naturaleza localizada de las funciones de nodo, otros algoritmos más simples han sido desarrollados para la formación de redes de base radial.

### 6.2.3 Los $K$-vecinos más cercanos regresivos

El método de regresión $K$-vecinos más cercanos (KNN, de las voces inglesas *K Nearest Neighbor Regression*) es un método no paramétrico que basa su predicción en las salidas previstas de los $K$-vecinos más cercanos de una consulta dada (Hastie *et al.*, 2005). Específicamente, dado un dato concreto se calcula la distancia euclídea entre dicho dato y todos los datos en el conjunto de entrenamiento. A continuación, se recogen los $K$-datos más cercanos de entrenamiento y se establece la predicción como la media de los valores de las salidas previstas para estos $K$-datos. Hablando cuantitativamente, dado $\mathscr{J}(x)$ el conjunto de $K$-vecinos más cercanos del dato $x$. Entonces, la predicción viene dada por la ecuación 6.3.

$$\hat{y} = \frac{1}{K} \sum_{m \in \mathscr{J}(x)} y_m \tag{6.3}$$

donde $y_m$ es la salida prevista del dato de entrenamiento $x_m$. Naturalmente, $K$ es un parámetro clave en este método y tiene que ser seleccionado con cuidado. Un valor alto de $K$ dará lugar a un ajuste más uniforme y por lo tanto, a una variación menor a expensas de un mayor sesgo. Viceversa para un valor pequeño de $K$.

[1] La función de activación de un nodo define la salida de un nodo dada una entrada o un conjunto de entradas.

### 6.2.4 El soporte vectorial regresivo

La regresión de vectores soporte (SVR, de los términos ingleses *Support Vector Regression*) (Schölkopf y Smola, 2002; Smola y Schölkopf, 2004) es un método competente basado en el uso de un espacio multidimensional de características (formado mediante la transformación de las variables originales) y en la penalización de la complejidad resultante empleando un término de penalización añadido a la función de error. Considerando en primer lugar un modelo lineal, entonces, la predicción viene dada por la ecuación 6.4.

$$f(x) = w^T x + b \tag{6.4}$$

donde $w$ es el vector de pesos, $b$ es el sesgo y $x$ es el vector de entrada. Dado $x_m$ e $y_m$ se denota respectivamente el vector de entrada de entrenamiento $m^t h$ y la salida prevista $m = 1,..., M$. La función de error viene dada por la ecuación 6.5.

$$J = \frac{1}{2}||w||^2 + C\sum_{m=1}^{M}|y_m - f(x_m)|_\epsilon \tag{6.5}$$

El primer término en la función de error es un término que penaliza la complejidad del modelo. El segundo término es la función de pérdida $\epsilon$-insensible[1], definida como $|y_m - f(x_m)|_\epsilon = \max\{0, |y_m - f(x_m)| - \epsilon\}$. Esta función no penaliza los errores por debajo de $\epsilon$, permitiendo un margen de maniobra a los parámetros para moverse con el fin de reducir la complejidad del modelo. Se puede demostrar que la solución que minimiza la función de error viene dada por la ecuación 6.6.

$$f(x) = \sum_{m=1}^{M}(\alpha_m^* - \alpha_m)x_m^T x + b \tag{6.6}$$

donde $\alpha_m$ y $\alpha_m^*$ son los multiplicadores de *Lagrange*[2]. Los vectores de entrenamiento que ofrecen multiplicadores de *Lagrange* distintos de cero son denominados *vectores soporte*, siendo éste un concepto clave en la teoría SVR. Los vectores no-soporte no contribuyen directamente a la solución, mientras que el número de vectores soporte es una cierta medida sobre el grado de complejidad

[1] La función de pérdida $\epsilon$-insensible es una aproximación de la función de pérdida de *Huber*, que habilita el poder obtener un conjunto disperso de vectores soporte.

[2] El método de los multiplicadores de *Lagrange* es un procedimiento para encontrar los máximos y los mínimos de funciones de varias variables sujetas a restricciones.

del modelo (Chalimourda *et al.*, 2004; Cherkassky y Ma, 2004). Este modelo se extiende al caso no lineal mediante el concepto del núcleo $\mathscr{K}$, dando una solución mostrada en la ecuación 6.7.

$$f(x) = \sum_{m=1}^{M} (\alpha_m^* - \alpha_m) \mathscr{K}(x_m^T x) + b \tag{6.7}$$

Un núcleo comúnmente usado es el núcleo *gaussiano*, asumiendo que su ancho es $\sigma_K$ (la desviación estándar de la función de *Gauss*[1]).

### 6.2.5 Los procesos *gaussianos*

La regresión de los procesos *gaussianos* (GP, del inglés *Gaussian Processes*) es un método no parámetrico basado en la modelización de las respuestas observadas de los diferentes datos de entrenamiento (valores de la función) como una variable normal aleatoria multivariante (Rasmussen, 2006). Para estos valores de la función en una distribución a *priori* se asume que garantizan las propiedades de uniformidad de la función. Específicamente, si los vectores de entrada están próximos entre sí (desde el punto de vista de la distancia euclídea), la correlación entre los 2 valores de la función es alta y decrece a medida que ambos vectores se alejan el uno del otro. La distribución a *posteriori* de un valor de la función a predecir puede ser obtenida empleando la distribución asumida a *priori* mediante la aplicación de manipulaciones simples probabilísticas.

Dado $V(X,X)$ denota la matriz de covarianza entre los valores de la función, donde $X$ es la matriz de los vectores de entrada del conjunto de datos de entrenamiento y dado el elemento $(i,j)$ de $V(X,X)$ siendo $V(x_i,x_j)$, donde $x_i$ denota el vector de entrada de entrenamiento $i$. Una matriz típica de covarianza se muestra en la ecuación 6.8.

$$V(x_i,x_j) = \sigma_f^2 e^{-\frac{||x_i - x_j||^2}{2\gamma^2}} \tag{6.8}$$

En consecuencia, el vector de los valores de la función $f$ (donde $f_i$ es el valor de la función para el valor de entrenamiento $i$) obedece la siguiente densidad de *Gauss* multivariante, mostrada en la ecuación 6.9.

[1]La función de *Gauss* o distribución normal es una de las distribuciones de probabilidad de variable continua empleada con mayor frecuencia en la vida real, ya que permite modelar numerosos fenómenos naturales, sociales y psicológicos.

$$f \sim \mathcal{N}_f(0, V(X,X)) \tag{6.9}$$

donde $\mathcal{N}_f(\mu, \Sigma)$ denota una función multivariante de densidad normal en la variable $f$ con media $\mu$ y matriz de covarianza $\Sigma$.

Además, algunos valores independientes de media cero producen ruido, teniendo también una desviación estándar $\sigma_n$. Por ello, se supone que se añaden a los valores de la función para producir las respuestas deseadas (los valores previstos). Como por ejemplo, la ecuación 6.10.

$$y = f + \epsilon \tag{6.10}$$

donde $y$ es el vector de las salidas previstas y $\epsilon$ es el vector de ruido aditivo, cuyos componentes se supone que son independientes. Los términos $f_i$ representan los valores de la función inherente y éstos son los valores que nos gustaría predecir para los datos de prueba.

Entonces, para un vector de entrada $x_*$ dado, la predicción $\hat{f}_*$ se deriva como la ecuación 6.11.

$$\hat{f}_* = E(f_*|X, y, x_*) = V(x_*, X)[V(X,X) + \sigma_n^2 I]^{-1} y \tag{6.11}$$

La ecuación 6.11 se obtiene mediante manipulaciones estándar empleando la regla de *Bayes* aplicada sobre las densidades normales dadas.

## 6.3 Validación empírica

Hemos clasificado los conjuntos de datos en *Tipo de Comentario* y *Grado del Comentario*, empleando las mismas clases que en el conjunto de comentarios inicial (explicadas en la sección 3.2.3.1). A continuación se desarrollan las 2 categorías de clasificación:

- **Tipo de comentario.** El tipo de comentario indica qué clase de información y de qué tipo realiza el usuario. A su vez se divide en:
  - *Aporte.* El usuario contribuye añadiendo nueva información.
  - *Irrelevante.* Estos comentarios no contribuyen al artículo principal ni a otros comentarios anteriores.

  - *Opinión.* Estos comentarios expresan la opinión particular del usuario sobre un tema tratado en la noticia o en otros comentarios.

- **Grado del comentario.** Hemos categorizado el nivel de controversia en 4 grados diferentes: normal, polémico, muy polémico y, también, un atributo adicional que se utiliza para comentarios graciosos o irónicos.

  - *Normal.* Un comentario normal es aquel que no plantea ninguna controversia o ironía. No busca hacer daño ni ser hiriente.
  - *Polémico.* Un comentario que, a propósito, busca la polémica con un tono perjudicial pero no de una manera exagerada.
  - *MuyPolémico.* Busca crear controversia en un modo exagerado, siendo perjudicial o irrespetuoso. Es lo conocido como usuario trol.
  - *Gracioso.* Estos comentarios intentar ser graciosos o tratan de hacer alguna broma, incluso irónicamente.

A continuación, la Figura 6.2 muestra un extracto del archivo de un usuario formado por el título, el resumen, las etiquetas de la noticia, las categorías, los comentarios que el usuario publicó en la noticia y la clase formada por las etiquetas realizadas manualmente de cada uno de los comentarios con el siguiente formato:

*C: TipoComentario-GradoComentario*

Para poder obtener resultados y analizarlos mediante las técnicas de predicción y análisis de series temporales, es necesario que los conjuntos de datos que se vayan a procesar estén formados por valores numéricos. Con el fin de obtener unos conjuntos de datos numéricos, hemos convertido el valor de las clases etiquetadas, y ya clasificadas, a un nuevo atributo numérico. Gracias a este nuevo atributo, hemos podido representar estas etiquetas en una serie temporal de datos numéricos para aplicar los métodos de series temporales sobre dicho conjunto. La conversión realizada se muestra en la Tabla 6.2.

Gracias a la conversión realizada, dispondremos de 2 nuevos atributos totalmente numéricos, los cuales podrán ser procesados por los diferentes modelos predictivos de series temporales explicados anteriormente: (i) *claseOpinionAporteIrrelevante* y (ii) *claseNormalPolemicoMuyPolemico*, ambos dentro de un rango de valores entre [0, 2].

TITULO: Un joven presencia el asesinato de su novia a través de una webcam
RESUMEN: Una joven china ha sido asesinada en la Universidad de York en Toronto mientras hablaba con su novio a través de una cámara web. Su pareja, residente en China, ha presenciado como un desconocido entraba en la habitación de la víctima y forcejeaba con ella. Relacionada con esta noticia en inglés: www.meneame.net/story/estudiante-china-muerta-canada-despues-novio-vie

etiquetas: joven, asesinato, novia, webcam
CATEGORIAS: actualidad, sucesos

COMENTARIO: #8 Yo ya había dicho que era igual que la otra noticia pero en español, tampoco es para que os ensañeis a votar negativamente como duplicada y encima cuando ya no se puede quitar la noticia 6 votos: 0 el 20-04-2011 13:40 UTC por alfonsoblog Num Respuestas: 0
C: Irrelevante-Normal

COMENTARIO: #6 #5 Si bueno, lo único es que aquí está la información en castellano, queria subirla para quien le fuera más cómodo leerla así 6 votos: 0 el 20-04-2011 13:10 UTC por alfonsoblog Num Respuestas: 0
C: Irrelevante-Normal

FIN-NOTICIA

**Figura 6.2:** Extracto del archivo de un usuario etiquetado.

### 6.3.1 **Metodología**

Una vez formado los conjuntos de datos a procesar en archivos ARFF (en este caso 13 archivos, uno por cada usuario seleccionado), hemos evaluado la precisión de nuestro método propuesto. A continuación, hemos efectuado la siguiente metodología:

**Tabla 6.2:** Conversión de las etiquetas de los conjuntos de datos a valores numéricos.

| Valor numérico | Tipo de comentario | Grado del comentario |
|---|---|---|
| 0 | Opinión | Normal/Gracioso |
| 1 | Irrelevante | Polémico |
| 2 | Aporte | Muy Polémico |

- **Número de unidades temporales.** Este parámetro controla la cantidad de unidades temporales futuras que el sistema será capaz de predecir. En este caso, únicamente se realizará una predicción de 1 salto. Es decir, solamente prediremos el siguiente comentario del usuario.

- **Etiqueta temporal.** Su función es definir el valor temporal en el cual se divide la serie. En esta investigación, la series temporales disponen del atributo fecha. Este atributo es aleatorio y ordenado de manera ascendente, pero sin ningún rigor de tipo temporal entre los valores o comentarios (no son diarios, mensuales o anuales, sino que un comentario puede ser publicado en cualquier momento de cualquier fecha).

  Al obviar este valor, únicamente definiremos como etiqueta temporal el número del comentario correspondiente. Mientras que si definimos el atributo fecha como la etiqueta temporal, existirá cierta probabilidad de pronosticar la fecha exacta de la siguiente aparición o comentario.

- **Variables rezagadas.** Principal mecanismo por el que la relación entre los valores actuales y los valores pasados de una serie temporal pueden ser recogidos por los algoritmos de aprendizaje automático. Crean una *ventana* o *instantánea* sobre un periodo de tiempo. Esencialmente, el número de variables rezagadas creadas determina el tamaño de la *ventana*.

  El valor mínimo de la variable rezagada que emplearemos será 1. Este valor nos indicará el mínimo salto previo que se ejecutará para crear la variable rezagada. En cuanto al valor máximo de la variable rezagada, este será de 2. Nos indicará el salto previo máximo a realizar para crear la variable rezagada. Es decir, aparte de los valores de la propia serie temporal, estaremos modelando la influencia de los 2 últimos comentarios que el usuario haya publicado.

- **Aprendizaje del modelo.** Para cada conjunto de datos, hemos efectuado la etapa de aprendizaje de cada algoritmo utilizando diferentes pará-

metros o distintos tipos de algoritmos de aprendizaje, dependiendo del modelo específico. Los algoritmos usan los parámetros por defecto en la herramienta de aprendizaje automático WEKA (acrónimo del inglés *Waikato Environment for Knowledge*). En particular, hemos usado los siguiente modelos:

  - *Perceptrón multicapa (MLP, de las voces inglesas Multilayer Perceptron)* con sus valores por defecto.
  - *Redes neuronales basadas en funciones de base radial (RBF, de las voces inglesas Radial Basis Function Neural Network)* con sus valores por defecto.
  - *K-vecinos más cercanos regresivos (KNN, de las voces inglesas K Nearest Neighbor Regression)*. Hemos realizado experimentos con diferentes valores de $K$, $K = 1$, $K = 2$, $K = 3$, $K = 4$ y $K = 5$.
  - *Regresión de vectores soporte (SVR, de los términos ingleses Support Vector Regression)*. Hemos elaborado experimentos con un núcleo polinomial (del inglés *polynomial kernel*) (Amari y Wu, 1999), un núcleo polinomial normalizado (del inglés *normalized polynomial kernel*) (Maji *et al.*, 2008), un núcleo *Pearson VII* (Üstün *et al.*, 2007) y un núcleo basado en funciones de base radial (RBF, del inglés *radial basis function*) (Cho *et al.*, 2008).
  - *Procesos gaussianos (GP, del inglés Gaussian Processes)*. Al igual que en la regresión de vectores soporte, hemos experimentado con un núcleo polinomial, un núcleo polinomial normalizado, un núcleo *Pearson VII* y un núcleo basado en funciones de base radial.

- **Evaluación del modelo.** Para evaluar el modelo temporal, hemos separado el conjunto de datos en 2 subconjuntos. Por una parte, un subconjunto con el 70 % de los datos para poder realizar la fase de entrenamiento y por otra parte, un subconjunto con el 30 % de los datos con el fin de ejecutar la fase de prueba.

  Debido a que disponemos de una serie de observaciones $X_t$ y de predicciones $Y_t$ para $n$ periodos temporales, obtendremos $n$ términos de error, los cuales serán recogidos mediante las siguientes medidas estadísticas que nos indican las tasas error correspondientes:

  - *Error absoluto medio (MAE, del inglés Mean Absolute Error)*. Definido por Stauffer y Seaman (1990), mide la magnitud media de los errores en un conjunto de predicciones, sin considerar sus direcciones.

Es decir, mide la precisión para las variables continuas. La ecuación 6.12 muestra la forma matemática de dicha tasa de error, que desarrollada hace referencia al promedio de la muestra de verificación de los valores absolutos de las diferencias entre la predicción $Y_t$ y la observación correspondiente $X_t$. El MAE es una tasa lineal, la cual indica que todas las diferencias individuales son ponderadas equitativamente en el promedio.

$$MAE(X,Y) = \sum_{t=1}^{n} \frac{|Y_t - X_t|}{n} \tag{6.12}$$

– *Raíz del error cuadrático medio (RMSE, de las voces inglesas Root Mean Squared Error)*. Definida por Pielke (2002), es una regla de tasa cuadrática que mide la magnitud media del error. La ecuación 6.13 muestra el RMSE matemáticamente hablando. Expresando dicha fórmula, la diferencia entre los valores de predicción $Y_t$ y los correspondientes valores observados $X_t$ son elevados al cuadrado y a continuación, promediados sobre la muestra. Finalmente, se calcula la raíz cuadrada del promedio. Ya que los errores son elevados al cuadrado antes de ser promediados, el RMSE da un peso relativamente alto a los grandes errores, esto significa que el RMSE es más útil cuando los errores elevados son particularmente no deseados.

$$RMSE(X,Y) = \frac{1}{n}\sqrt{\sum_{t=1}^{n}(Y_t - X_t)^2} \tag{6.13}$$

El MAE y el RMSE pueden ser usados conjuntamente para diagnosticar la variación de los errores en un conjunto de predicciones. El valor de RMSE será siempre mayor o igual que el valor de MAE. A mayor diferencia entre ellos, mayor será la variación en los errores individuales de la muestra. Si el valor de RMSE es igual al valor de MAE, entonces todos los errores son de la misma magnitud.

Tanto los valores de MAE como de RMSE pueden variar de 0 a $\infty$. Son tasas orientadas negativamente; es decir, los valores más bajos son los mejores.

## 6.4 Discusión de los resultados

Esta sección expone los resultados obtenidos siguiendo la metodología realizada. Hemos diferenciado los resultados teniendo en cuenta la serie temporal a la que pertenecen: (i) *OpinionAporteIrrelevante* o (ii)*NormalPolemicoMuyPolemico*.

### 6.4.1 Serie temporal *OpinionAporteIrrelevante*

A continuación, esta sección muestra los resultados logrados para la clasificación referente al tipo de comentario: (i) Opinión, (ii) Aporte o (iii) Irrelevante. Dicha sección expone los valores obtenidos tanto por el MAE como por el RMSE, así como el resultado de la predicción.

**Tabla 6.3:** Resultados obtenidos para MAE y RMSE, en las fases de entrenamiento y prueba, sobre la categorización *OpinionAporteIrrelevante* para el usuario *gabi10*.

| | | Entrenamiento (N=143) | | Prueba (N=62) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,5456 | 0,5215 | 0,5662 | 0,5304 | 0,5775 |
| GP: Polinomial | 0,6074 | 0,5218 | 0,5669 | 0,5395 | 0,5888 |
| GP: PVK | 0,5285 | 0,5074 | 0,5492 | 0,5441 | 0,6039 |
| GP: RBF | 0,5709 | 0,5301 | 0,567 | 0,5441 | 0,5851 |
| MLP | 0,5965 | 0,5154 | 0,5621 | 0,5539 | 0,621 |
| Redes RBF | 0,6456 | 0,5267 | 0,5674 | 0,5484 | 0,5902 |
| SVR: Polinomial normal. | 0,0024 | 0,4631 | 0,6976 | 0,4494 | 0,707 |
| SVR: Polinomial | 0,1441 | 0,4776 | 0,6847 | 0,507 | 0,7494 |
| SVR: PVK | 0,0033 | 0,4408 | 0,6932 | 0,4842 | 0,7607 |
| SVR: RBF | 0,9635 | 0,5022 | 0,691 | 0,5314 | 0,7141 |
| KNN K = 1 | 0,5172 | 0,4991 | 0,5452 | 0,5541 | 0,6402 |
| KNN K = 2 | 0,5172 | 0,5111 | 0,5565 | 0,5338 | 0,5884 |
| KNN K = 3 | 0,5172 | 0,5111 | 0,5565 | 0,5338 | 0,5884 |
| KNN K = 4 | 0,5172 | 0,5138 | 0,5598 | 0,5344 | 0,5824 |
| KNN K = 5 | 0,5172 | 0,5138 | 0,5598 | 0,5344 | 0,5824 |

La Tabla 6.3 presenta los resultados del usuario *gabi10* en cuanto a la clasificación *OpinionAporteIrrelevante*. En este caso, el clasificador SVR con un núcleo polinomial normalizado devuelve los mejores valores de MAE en la fase de prueba: 0,4494. Siendo el clasificador que mejor rendimiento aporta, su valor de predicción es de 0,0024. Es decir, el próximo comentario a publicar por *gabi10* será de tipo Opinión. Este valor difiere de los obtenidos con el resto de clasificadores, ya que en su mayoría ofrecen valores superiores a 0,5, con lo que podríamos clasificar el comentario como Irrelevante en vez de Opinión.

**Tabla 6.4:** Resultados logrados en cuanto a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, respecto a la clasificación *OpinionAporteIrrelevante* para el usuario *pavlenka*.

| | | Entrenamiento (N=151) | | Prueba (N=65) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,3558 | 0,4577 | 0,4908 | 0,4939 | 0,5341 |
| GP: Polinomial | 0,3905 | 0,4645 | 0,4923 | 0,4956 | 0,5319 |
| GP: PVK | 0,3663 | 0,4547 | 0,4882 | 0,497 | 0,5359 |
| GP: RBF | 0,3583 | 0,4625 | 0,4927 | 0,4947 | 0,5346 |
| MLP | 0,5983 | 0,5222 | 0,5312 | 0,5232 | 0,5379 |
| Redes RBF | 0,4121 | 0,4592 | 0,4914 | 0,495 | 0,5334 |
| SVR: Polinomial normal. | 0,0001 | 0,3577 | 0,6087 | 0,4462 | 0,6903 |
| SVR: Polinomial | 0 | 0,3576 | 0,609 | 0,4462 | 0,6906 |
| SVR: PVK | 0,0013 | 0,3513 | 0,6028 | 0,4526 | 0,6911 |
| SVR: RBF | 0,0029 | 0,3583 | 0,6076 | 0,4466 | 0,6893 |
| KNN K = 1 | 0,3714 | 0,4512 | 0,4868 | 0,4953 | 0,5338 |
| KNN K = 2 | 0,3714 | 0,4574 | 0,4898 | 0,4953 | 0,5338 |
| KNN K = 3 | 0,3714 | 0,4574 | 0,4898 | 0,4953 | 0,5338 |
| KNN K = 4 | 0,3714 | 0,4574 | 0,4898 | 0,4953 | 0,5338 |
| KNN K = 5 | 0,3714 | 0,4574 | 0,4898 | 0,4953 | 0,5338 |

La Tabla 6.4 ofrece los resultados para la categoría *OpinionAporteIrrelevante* en referencia al usuario *pavlenka*. El modelo de predicción que mejor rendimiento obtiene es el SVR con núcleo polinomial normalizado, alcanzado un valor de MAE de 0,4462 en la fase de prueba y de 0,3577 en la fase de entrenamiento. Por su parte, dicho clasificador ofrece como predicción el valor 0,0001 en el siguiente salto en nuestra serie temporal. Es decir, el siguiente comentario que realizará el usuario *pavlenka* será de tipo Opinión. En esta clasificación, la mayor parte de los modelos de predicción han logrado valores cercanos a 0, con lo que la mayoría de ellos predice que el siguiente comentario será de tipo Opinión.

Por otra parte, queremos destacar el rendimiento, tanto en fase de entrenamiento como en fase de prueba, de los clasificadores $K$-vecinos más cercanos regresivos y de los procesos *gaussianos*. Ambos modelos de predicción ofrecen valores considerables en cuanto a la tasa de error MAE (0,4574 y 0,4908 respectivamente en la fase de entrenamiento; 0,4953 y 0,4939 respectivamente en la fase de prueba) y una variación mínima en la tasa de error RMSE, respecto a la tasa de error MAE (0,4898 y 0,4908 respectivamente en la fase de entrenamiento; 0,5338 y 0,5341 respectivamente en la fase de prueba).

**Tabla 6.5:** Resultados obtenidos en base a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, sobre la categorización *OpinionAporteIrrelevante* para el usuario *batis*.

| | | Entrenamiento (N=261) | | Prueba (N=112) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,6215 | 0,5007 | 0,5116 | 0,5159 | 0,5432 |
| GP: Polinomial | 0,5806 | 0,5025 | 0,5129 | 0,5122 | 0,5393 |
| GP: PVK | 0,6165 | 0,4985 | 0,5094 | 0,5176 | 0,5449 |
| GP: RBF | 0,5487 | 0,5048 | 0,5126 | 0,5183 | 0,5426 |
| MLP | 0,7317 | 0,4966 | 0,535 | 0,4777 | 0,534 |
| Redes RBF | 0,5533 | 0,5056 | 0,5137 | 0,5189 | 0,5427 |
| SVR: Polinomial normal. | 1 | 0,4552 | 0,6736 | 0,5196 | 0,73 |
| SVR: Polinomial | 0,998 | 0,4599 | 0,6823 | 0,5356 | 0,7544 |
| SVR: PVK | 0,9987 | 0,4523 | 0,6703 | 0,5116 | 0,7231 |
| SVR: RBF | 0,9979 | 0,4882 | 0,6726 | 0,4123 | 0,6205 |
| KNN K = 1 | 0,6133 | 0,496 | 0,5086 | 0,515 | 0,5478 |
| KNN K = 2 | 0,6133 | 0,4993 | 0,51 | 0,5177 | 0,5444 |
| KNN K = 3 | 0,6133 | 0,4993 | 0,5101 | 0,5177 | 0,5446 |
| KNN K = 4 | 0,6133 | 0,4993 | 0,5101 | 0,5177 | 0,5446 |
| KNN K = 5 | 0,6133 | 0,4993 | 0,5101 | 0,5177 | 0,5446 |

La Tabla 6.5 muestra los valores que los modelos de predicción seleccionados devuelven para la clasificación de la serie temporal *OpiniónAporteIrrelevante* respecto al usuario *batis*. El algoritmo que mejor valor de tasa de error absoluto medio proporciona es el SVR con un núcleo basado en funciones de base radial: 0,4123 en la fase de prueba y 0,4882 en la fase de entrenamiento. En el mismo sentido, este clasificador predice como siguiente comentario del usuario *batis* un comentario de tipo Irrelevante, ya que el valor de la predicción es de 0,9979. En este caso, la mayor parte de los clasificadores ofrecen valores de predicción mayores a 0,5, con lo que todos ellos respaldan el tipo de comentario previsto por el clasificador SVR con núcleo RBF.

En cuanto el resto de modelos predictivos, los valores obtenidos para las 2 tasas de error (MAE y RMSE) son similares entre ellos, tanto para la fase de entrenamiento como para la fase de prueba. Respecto a la salida predictiva que ofrecen esta clase de modelos, también se observa una homogeneidad en los resultados, obteniendo mayores valores en los clasificadores SVR. Particularmente, destacan los algoritmos KNN, ya que a partir del empleo de 2 vecinos ($K = 2$, $K = 3$, $K = 4$ y $K = 5$) los resultados logrados por el clasificador son prácticamente idénticos.

**Tabla 6.6:** Resultados logrados en cuanto a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, respecto a la clasificación *OpinionAporteIrrelevante* para el usuario *hangla*.

| | | Entrenamiento (N=242) | | Prueba (N=105) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,4254 | 0,4896 | 0,5005 | 0,5029 | 0,5073 |
| GP: Polinomial | 0,4602 | 0,4921 | 0,5009 | 0,506 | 0,5087 |
| GP: PVK | 0,4167 | 0,484 | 0,4983 | 0,5011 | 0,5072 |
| GP: RBF | 0,424 | 0,4893 | 0,5011 | 0,5149 | 0,5204 |
| MLP | 0,1643 | 0,4407 | 0,5559 | 0,5595 | 0,6366 |
| Redes RBF | 0,3998 | 0,4896 | 0,5017 | 0,5131 | 0,5185 |
| SVR: Polinomial normal. | 0,0033 | 0,4219 | 0,6538 | 0,6183 | 0,7846 |
| SVR: Polinomial | 0,002 | 0,4218 | 0,6543 | 0,6186 | 0,7852 |
| SVR: PVK | 0,0029 | 0,4134 | 0,6114 | 0,5207 | 0,6434 |
| SVR: RBF | 0,0038 | 0,4217 | 0,6533 | 0,6172 | 0,7827 |
| KNN K = 1 | 0,4151 | 0,4818 | 0,4976 | 0,5012 | 0,507 |
| KNN K = 2 | 0,4151 | 0,4862 | 0,4999 | 0,5012 | 0,507 |
| KNN K = 3 | 0,4151 | 0,4862 | 0,4999 | 0,5012 | 0,507 |
| KNN K = 4 | 0,4151 | 0,4862 | 0,4999 | 0,5012 | 0,507 |
| KNN K = 5 | 0,4151 | 0,4862 | 0,4999 | 0,5012 | 0,507 |

La Tabla 6.6 expone los resultados obtenidos por los clasificadores de predicción, en cuanto a la categorización *OpinionAporteIrrelevante* se refiere. En esta ocasión, el usuario analizado responde al nombre de *hangla*. El mejor algoritmo predictivo ha sido el proceso *gaussiano* con núcleo *Pearson VII*, devolviendo un valor de 0,5011 de tasa de error MAE en la fase de prueba y de 0,484 de MAE en la fase de entrenamiento. En referencia a la predicción del siguiente comentario, el valor obtenido (0,4167) se acerca más a un comentario de tipo Opinión (valor 0) que a un comentario de tipo Irrelevante (valor 1).

Respecto al comportamiento del resto de modelos predictivos, excepto los algoritmos de soporte vectorial regresivo y el perceptrón multicapa, todos logran una salida prácticamente igual en torno al valor 0,41. Por otra parte, tanto los procesos *gaussianos* como los *K*-vecinos más cercanos regresivos obtienen los mejores resultados respecto a las variaciones en la tasa del error absoluto medio y la tasa de la raíz del error cuadrático medio, tanto en la fase de entrenamiento como en la fase de prueba. Por el contrario, los clasificadores SVR obtienen valores elevados de MAE, así como de RMSE (únicamente en la fase de prueba) con lo que existe un aumento en la variación entre las muestras individuales.

**Tabla 6.7:** Resultados obtenidos en base a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, sobre la categorización *OpinionAporteIrrelevante* para el usuario *willies*.

| | | Entrenamiento (N=140) | | Prueba (N=61) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,5289 | 0,499 | 0,4997 | 0,5031 | 0,5037 |
| GP: Polinomial | 0,5204 | 0,499 | 0,4996 | 0,502 | 0,5026 |
| GP: PVK | 0,5173 | 0,4983 | 0,4991 | 0,5035 | 0,5043 |
| GP: RBF | 0,5123 | 0,4992 | 0,4994 | 0,5016 | 0,5018 |
| MLP | 0,6862 | 0,4943 | 0,524 | 0,5172 | 0,5446 |
| Redes RBF | 0,5065 | 0,499 | 0,4995 | 0,5009 | 0,5013 |
| SVR: Polinomial normal. | 0,9982 | 0,4715 | 0,6858 | 0,5898 | 0,7672 |
| SVR: Polinomial | 0,5177 | 0,4823 | 0,6023 | 0,5015 | 0,6053 |
| SVR: PVK | 0,9989 | 0,4715 | 0,6859 | 0,59 | 0,7675 |
| SVR: RBF | 0,9908 | 0,4856 | 0,6929 | 0,5404 | 0,7309 |
| KNN K = 1 | 0,5152 | 0,4982 | 0,4991 | 0,5034 | 0,5043 |
| KNN K = 2 | 0,5152 | 0,4982 | 0,4991 | 0,5034 | 0,5043 |
| KNN K = 3 | 0,5152 | 0,4982 | 0,4991 | 0,5034 | 0,5043 |
| KNN K = 4 | 0,5152 | 0,4982 | 0,4991 | 0,5034 | 0,5043 |
| KNN K = 5 | 0,5152 | 0,4982 | 0,4991 | 0,5034 | 0,5043 |

La Tabla 6.7 hace referencia a los resultados recogidos del usuario *willies* respecto a la clasificación *OpinionAporteIrrelevante*. En el caso de este usuario, el modelo de predicción basado en las redes neuronales basadas en funciones de base radial alcanza el valor más óptimo: 0,5009 en la fase de entrenamiento y 0,499 en la fase de prueba, en cuanto a la tasa de error MAE se refiere. En este mismo sentido, este clasificador ofrece una salida de 0,5065, en términos de predicción, lo cual podría clasificarse tanto como comentario de tipo Opinión, como comentario de tipo Irrelevante. Sin embargo, observando el comportamiento del resto de clasificadores, los cuales todos ellos se acercan más al comentario de tipo Irrelevante, podemos instaurar el comentario tipo Irrelevante como el próximo comentario a publicar por el usuario *willies*.

Respecto al resto de clasificadores predictivos, cabe destacar el rendimiento ofrecido tanto por los procesos *gaussianos* como por los *K*-vecinos más cercanos regresivos, obteniendo valores similares de MAE y RMSE tanto en la fase de entrenamiento como en la fase de prueba. Por otra parte, existe una gran diferencia de valores entre el MAE y el RMSE en los algoritmos SVR, lo cual nos indica un gran aumento en la variación entre los datos de la muestra.

**Tabla 6.8:** Resultados logrados en cuanto a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, respecto a la clasificación *OpinionAporteIrrelevante* para el usuario *tachyon*.

| | | Entrenamiento (N=249) | | Prueba (N=108) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,6395 | 0,4647 | 0,492 | 0,4611 | 0,4892 |
| GP: Polinomial | 0,6331 | 0,4662 | 0,4918 | 0,4627 | 0,4891 |
| GP: PVK | 0,6449 | 0,4539 | 0,486 | 0,4575 | 0,4899 |
| GP: RBF | 0,6459 | 0,4625 | 0,4916 | 0,4596 | 0,4895 |
| MLP | 0,3963 | 0,5368 | 0,5532 | 0,5364 | 0,5541 |
| Redes RBF | 0,6389 | 0,4615 | 0,4915 | 0,4587 | 0,4894 |
| SVR: Polinomial normal. | 0,9963 | 0,3623 | 0,5994 | 0,3528 | 0,5914 |
| SVR: Polinomial | 0,9974 | 0,3619 | 0,6002 | 0,3523 | 0,5921 |
| SVR: PVK | 0,9966 | 0,3461 | 0,5864 | 0,3567 | 0,5934 |
| SVR: RBF | 0,9976 | 0,3618 | 0,6003 | 0,3522 | 0,5922 |
| KNN K = 1 | 0,6505 | 0,4479 | 0,4845 | 0,4607 | 0,4967 |
| KNN K = 2 | 0,6505 | 0,4531 | 0,488 | 0,4546 | 0,4884 |
| KNN K = 3 | 0,6505 | 0,4581 | 0,4912 | 0,4546 | 0,4884 |
| KNN K = 4 | 0,6505 | 0,4581 | 0,4912 | 0,4546 | 0,4884 |
| KNN K = 5 | 0,6505 | 0,4581 | 0,4912 | 0,4546 | 0,4884 |

La Tabla 6.8 referencia a la categorización *OpinionAporteIrrelevante* perteneciente al usuario *tachyon*. El algoritmo que devuelve el mejor resultado en términos de MAE para esta clasificación es el SVR con un núcleo basado en funciones de base radial con un valor de 0,3522 en la fase de prueba y de 0,3618 en la fase de entrenamiento. Gracias a estos valores, podemos situar el próximo comentario del usuario *tachyon* como un comentario de tipo Irrelevante, debido a que la salida del clasificador devuelve un valor de 0,9976. Esta predicción es apoyada por el resto de algoritmos, ya que la mayoría de ellos se acerca al valor 1 (comentario tipo Irrelevante), excepto el perceptrón multicapa cuyo resultado es 0,3963.

En este caso, la mayoría de los modelos predictivos ofrecen valores de tasa de error notables (tanto MAE como RMSE). No obstante, las 4 variantes de núcleo empleadas en el clasificador SVR sobresalen respecto al resto, obteniendo valores en torno al 0,35 en la fase de prueba, alrededor del 0,36 en la fase de entrenamiento y cercanos 0,99 como salida del modelo de predicción. En este caso, los peores resultados se logran al categorizar mediante el algoritmo perceptrón multicapa, ya que devuelve unos valores de 0,53 de MAE y 0,55 de RMSE, tanto en la fase de entrenamiento como en la fase de prueba.

**Tabla 6.9:** Resultados obtenidos en base a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, sobre la categorización *OpinionAporteIrrelevante* para el usuario *berthax*.

| | | Entrenamiento (N=156) | | Prueba (N=67) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,5156 | 0,5092 | 0,5378 | 0,5436 | 0,5618 |
| GP: Polinomial | 0,5193 | 0,5128 | 0,5392 | 0,5385 | 0,5553 |
| GP: PVK | 0,5128 | 0,4987 | 0,5308 | 0,5467 | 0,5692 |
| GP: RBF | 0,5777 | 0,5133 | 0,5404 | 0,5352 | 0,5529 |
| MLP | 0,6308 | 0,5081 | 0,5415 | 0,5467 | 0,5707 |
| Redes RBF | 0,561 | 0,5136 | 0,541 | 0,532 | 0,5496 |
| SVR: Polinomial normal. | 0,998 | 0,4617 | 0,6783 | 0,6118 | 0,7814 |
| SVR: Polinomial | 0,9965 | 0,4617 | 0,6781 | 0,6119 | 0,7812 |
| SVR: PVK | 0,9987 | 0,4426 | 0,664 | 0,6116 | 0,7807 |
| SVR: RBF | 1 | 0,4616 | 0,6793 | 0,6119 | 0,7822 |
| KNN K = 1 | 0,5128 | 0,488 | 0,5279 | 0,543 | 0,5692 |
| KNN K = 2 | 0,5128 | 0,4915 | 0,5297 | 0,543 | 0,5692 |
| KNN K = 3 | 0,5128 | 0,5061 | 0,5378 | 0,5498 | 0,5719 |
| KNN K = 4 | 0,5128 | 0,5071 | 0,5382 | 0,5522 | 0,5736 |
| KNN K = 5 | 0,5128 | 0,5071 | 0,5382 | 0,5522 | 0,5736 |

La Tabla 6.9 presenta los resultados obtenidos a la hora de clasificar al usuario *berthax* en la categorización de tipo de comentario *OpinionAporteIrrelevante*. El modelo de predicción de redes neuronales basadas en funciones de base radial ofrece los mejores resultados para ambas tasas de error (MAE y RMSE) y para ambas fases (entrenamiento y prueba). En la fase de prueba, devuelve unos valores de 0,532 para la tasa de error MAE y 0,5496 para la tasa de error RMSE, mientras que en la fase de entrenamiento devuelve unos valores de 0,5136 para el MAE y 0,541 para el RMSE. En este caso, el próximo comentario a publicar por el usuario *berthax* será de tipo Irrelevante. Esto se debe a que el resultado en la predicción del mejor clasificador es de 0,561.

En esta clasificación de la serie temporal perteneciente al usuario *berthax*, sobresalen los altos porcentajes en la tasa de error absoluto medio por parte de los modelos de predicción SVR con valores en torno al 61 % y en la tasa raíz del error cuadrático medio con valores alrededor de 78 %, ambos resultados en la fase de prueba. Aunque por otra parte, estos mismos modelos obtienen los mejores valores de tasa de error en la predicción con resultados alrededor de 0,45 y uniformes en cuanto a la salida predictiva (llegando al valor 1).

**Tabla 6.10:** Resultados logrados en cuanto a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, respecto a la clasificación *OpinionAporteIrrelevante* para el usuario *hsc*.

| | | Entrenamiento (N=219) | | Prueba (N=94) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,6114 | 0,5203 | 0,5589 | 0,538 | 0,5887 |
| GP: Polinomial | 0,6127 | 0,5201 | 0,559 | 0,5371 | 0,5883 |
| GP: PVK | 0,621 | 0,5179 | 0,5566 | 0,5496 | 0,598 |
| GP: RBF | 0,5905 | 0,5268 | 0,5604 | 0,5378 | 0,5833 |
| MLP | 0,6309 | 0,5222 | 0,5602 | 0,5303 | 0,5819 |
| Redes RBF | 0,567 | 0,5279 | 0,5619 | 0,5369 | 0,5824 |
| SVR: Polinomial normal. | 0,9969 | 0,4732 | 0,6702 | 0,5712 | 0,7581 |
| SVR: Polinomial | 0,9988 | 0,4886 | 0,6981 | 0,4472 | 0,6678 |
| SVR: PVK | 0,9977 | 0,4614 | 0,704 | 0,6167 | 0,8234 |
| SVR: RBF | 0,9964 | 0,4886 | 0,6966 | 0,4479 | 0,667 |
| KNN K = 1 | 0,6275 | 0,5137 | 0,5558 | 0,5547 | 0,6026 |
| KNN K = 2 | 0,6275 | 0,518 | 0,5576 | 0,5547 | 0,6026 |
| KNN K = 3 | 0,6275 | 0,518 | 0,5577 | 0,5488 | 0,5984 |
| KNN K = 4 | 0,6275 | 0,518 | 0,5577 | 0,5488 | 0,5984 |
| KNN K = 5 | 0,6275 | 0,5187 | 0,5578 | 0,5471 | 0,5963 |

La Tabla 6.10 muestra los resultados logrados para la serie temporal del usuario *hsc* para la clasificación *OpinionAporteIrrelevante*. En esta categorización, el modelo de predicción que mejores resultados ha ofrecido es el SVR con núcleo polinomial. En términos de tasa de error MAE, obtiene un valor de 0,4472 en la fase de prueba y 0,4886 en la fase de entrenamiento. En términos de tasa de error RMSE, logra un valor de 0,6678 en la fase de prueba y 0,6981 en la fase de entrenamiento. Por otra parte, dicho clasificador consigue una salida predictiva de 0,9988, con lo cual el siguiente comentario a publicar por el usuario *hsc* es un comentario de tipo Irrelevante. En este sentido, todos los clasificadores obtiene valores que tienden al valor 1, con lo que todos los clasificadores predicen que el siguiente comentario de *hsc* será de tipo Irrelevante.

En referencia al comportamiento del resto de clasificadores de predicción, resalta el algoritmo SVR con núcleo *Pearson VII* con un valor de MAE de 61,67 % y un valor de RMSE de 82,34 %, en la fase de prueba. Sin embargo, en la fase de entrenamiento, obtiene el valor más bajo en términos de MAE con 46,14 %, mientras que obtiene el valor más elevado en términos de RMSE con 70,4 %. También destaca la uniformidad en la salida del valor de predicción en los algoritmos KNN para cualquier valor de $K \leq 5$.

**Tabla 6.11:** Resultados obtenidos en base a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, sobre la categorización *OpinionAporteIrrelevante* para el usuario *psalbendea*.

| | | Entrenamiento (N=212) | | Prueba (N=92) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,5 | 0,5182 | 0,5359 | 0,5121 | 0,5228 |
| GP: Polinomial | 0,5 | 0,5184 | 0,5362 | 0,5097 | 0,5209 |
| GP: PVK | 0,5028 | 0,5154 | 0,534 | 0,5126 | 0,5231 |
| GP: RBF | 0,5063 | 0,5188 | 0,5362 | 0,5111 | 0,5216 |
| MLP | 0,2293 | 0,5071 | 0,6085 | 0,4833 | 0,5755 |
| Redes RBF | 0,4992 | 0,5188 | 0,5364 | 0,5109 | 0,5213 |
| SVR: Polinomial normal. | 0,0032 | 0,4969 | 0,6979 | 0,4975 | 0,6736 |
| SVR: Polinomial | 0,002 | 0,5001 | 0,7319 | 0,4459 | 0,6824 |
| SVR: PVK | 0,0037 | 0,4813 | 0,7184 | 0,5006 | 0,7052 |
| SVR: RBF | 0 | 0,5001 | 0,7324 | 0,4458 | 0,6829 |
| KNN K = 1 | 0,5 | 0,5127 | 0,5333 | 0,514 | 0,5245 |
| KNN K = 2 | 0,5 | 0,5149 | 0,5343 | 0,514 | 0,5245 |
| KNN K = 3 | 0,5 | 0,5149 | 0,5343 | 0,514 | 0,5245 |
| KNN K = 4 | 0,5 | 0,5156 | 0,5346 | 0,5124 | 0,5226 |
| KNN K = 5 | 0,5 | 0,5181 | 0,5358 | 0,5124 | 0,5226 |

La Tabla 6.11 expone los valores obtenidos en referencia a la clasificación de la serie temporal *OpinionAporteIrrelevante* para el usuario *psalbendea*. En este caso, el modelo de predicción SVR con un núcleo basado en funciones de base radial logra el valor más bajo respecto a la tasa de error MAE, en la fase de prueba: 0,4458. Mientras que el valor obtenido respecto a la tasa de error RMSE es de 0,6829. En cambio, en la fase de entrenamiento, este mismo clasificador obtiene un valor de 0,5001 en términos de MAE y el valor más elevado de entre todos los modelos, en términos de RMSE: 0,7324. En este mismo sentido, el resultado predicho por el clasificador para el usuario *psalbendea* tiene un valor 0. Es decir, este usuario en su próxima publicación escribirá un comentario de tipo Opinión.

En cuanto al comportamiento del resto de algoritmos de predicción en esta serie temporal, resalta la heterogeneidad de resultados en referencia al valor del próximo comentario a publicar. Los procesos *gaussianos*, las redes neuronales basadas en funciones de base radial y los $K$-vecinos más cercanos regresivos alcanzan resultados cercanos al valor 0,5, mientras que el resto de los clasificadores (los SVR con distintos núcleos y el perceptrón multicapa) logran valores cercanos a 0, permitiendo una conversión fácil al tipo de comentario Opinión.

**Tabla 6.12:** Resultados logrados en cuanto a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, respecto a la clasificación *OpinionAporteIrrelevante* para el usuario *amadorcea*.

| | | Entrenamiento (N=74) | | Prueba (N=32) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,4868 | 0,5306 | 0,5697 | 0,5473 | 0,5935 |
| GP: Polinomial | 0,4868 | 0,5236 | 0,5637 | 0,5488 | 0,5903 |
| GP: PVK | 0,3936 | 0,5107 | 0,5492 | 0,5406 | 0,5821 |
| GP: RBF | 0,4733 | 0,536 | 0,5724 | 0,5626 | 0,6115 |
| MLP | 0,1838 | 0,4846 | 0,5967 | 0,5427 | 0,6368 |
| Redes RBF | 0,4146 | 0,5251 | 0,5698 | 0,5634 | 0,6107 |
| SVR: Polinomial normal. | 0,0029 | 0,4531 | 0,6663 | 0,5315 | 0,686 |
| SVR: Polinomial | 0,0019 | 0,4529 | 0,6335 | 0,5623 | 0,6841 |
| SVR: PVK | 0,0024 | 0,4464 | 0,6643 | 0,4814 | 0,6525 |
| SVR: RBF | 0,0016 | 0,494 | 0,7347 | 0,6167 | 0,8307 |
| KNN K = 1 | 0,4 | 0,5089 | 0,5448 | 0,54 | 0,594 |
| KNN K = 2 | 0,4 | 0,5134 | 0,5462 | 0,5505 | 0,5969 |
| KNN K = 3 | 0,4 | 0,5225 | 0,5631 | 0,5496 | 0,5962 |
| KNN K = 4 | 0,4 | 0,5243 | 0,5645 | 0,554 | 0,5992 |
| KNN K = 5 | 0,4 | 0,5243 | 0,5645 | 0,554 | 0,5992 |

En relación a la serie temporal categorizada como *OpinionAporteIrrelevante* perteneciente al usuario *amadorcea*, la Tabla 6.12 ofrece los resultados logrados por los diferentes tipos de clasificadores predictivos. El mejor comportamiento por parte de un algoritmo, haciendo referencia a la tasa de error MAE, es el SVR con núcleo *Pearson VII* obteniendo un valor de 48,14 % en la fase de prueba y 44,64 % en la fase de entrenamiento. En cuanto a la salida de dicho modelo SVR, muestra un valor de 0,0024 por lo que el siguiente comentario a publicar por el usuario *amadorcea* será de tipo Opinión. Este resultado está relacionado con el resto de resultados de los demás clasificadores, ya que en su totalidad mostraron resultados cercanos a este tipo de comentario.

Por otra parte, cabe mencionar tanto los clasificadores de procesos *gaussianos* como los algoritmos de $K$-vecinos más cercanos regresivos con todas las variante de $K$ (desde el valor 1 hasta el valor 5). Ambos conjuntos de modelos de predicción obtienen valores para la tasa de error MAE cercanos a 0,54 y valores para la tasa de error RMSE cercanos a 0,59, en la fase de prueba. Lo cual significa que la variación en los errores individuales de la serie temporal para el usuario *amadorcea* no es tan elevada con en el resto de algoritmos predictivos.

**Tabla 6.13:** Resultados obtenidos en base a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, sobre la categorización *OpinionAporteIrrelevante* para el usuario *anadelagua*.

| | | Entrenamiento (N=360) | | Prueba (N=155) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,5552 | 0,5033 | 0,5173 | 0,5054 | 0,5207 |
| GP: Polinomial | 0,5552 | 0,5034 | 0,5174 | 0,5051 | 0,5206 |
| GP: PVK | 0,5319 | 0,4999 | 0,5148 | 0,5084 | 0,524 |
| GP: RBF | 0,5486 | 0,5052 | 0,5179 | 0,5065 | 0,5207 |
| MLP | 0,8445 | 0,4772 | 0,5801 | 0,489 | 0,5966 |
| Redes RBF | 0,5148 | 0,5035 | 0,5173 | 0,5021 | 0,518 |
| SVR: Polinomial normal. | 0,9966 | 0,464 | 0,6796 | 0,4646 | 0,68 |
| SVR: Polinomial | 0,9988 | 0,464 | 0,6804 | 0,4646 | 0,6809 |
| SVR: PVK | 0,9988 | 0,4584 | 0,6763 | 0,4775 | 0,6901 |
| SVR: RBF | 0,9981 | 0,464 | 0,6798 | 0,4647 | 0,6803 |
| KNN K = 1 | 0,5352 | 0,4957 | 0,5136 | 0,5146 | 0,5341 |
| KNN K = 2 | 0,5352 | 0,4988 | 0,5154 | 0,5073 | 0,5241 |
| KNN K = 3 | 0,5352 | 0,4988 | 0,5154 | 0,5073 | 0,5241 |
| KNN K = 4 | 0,5352 | 0,5028 | 0,5171 | 0,5037 | 0,519 |
| KNN K = 5 | 0,5352 | 0,5028 | 0,5171 | 0,5037 | 0,519 |

La Tabla 6.13 hace referencia a los resultados ofrecidos por los diferentes algoritmos de predicción sobre la clasificación *OpinionAporteIrrelevante* para el usuario *anadelagua*. El modelo predictivo más óptimo para este usuario es el SVR con núcleo polinomial normalizado. Éste obtiene valores de 46,46 % para la tasa de error MAE y 68 % para la tasa de error RMSE, tanto para la fase de prueba como para la fase de entrenamiento. Respecto a la salida del clasificador, devuelve un valor de 0,9966 por lo que el usuario *anadelagua* la próxima vez que proceda a publicar un comentario, éste será de tipo Irrelevante.

Respecto al comportamiento en el resto de modelos de predicción, destaca la similitud en los valores obtenidos en la tasa de error MAE, con un máximo de 0,5146 y un mínimo de 0,4646; así como en la tasa de error RMSE, con un máximo de 0,6901 y un mínimo de 0,518, todos ellos en la fase de prueba. Por su parte, en la fase de entrenamiento, los clasificadores vuelven a obtener valores similares en ambas tasas de error (MAE y RMSE). Por último, los algoritmos de predicción vuelven a asemejarse en cuanto a la salida que ofrecen se refiere. Todos ellos predicen que el próximo comentario será de valor 1 o cercano a él, es decir que será un comentario de tipo Irrelevante.

**Tabla 6.14:** Resultados logrados en cuanto a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, respecto a la clasificación *OpinionAporteIrrelevante* para el usuario *alfonsoblog*.

| | | Entrenamiento (N=88) | | Prueba (N=39) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,6466 | 0,5056 | 0,5359 | 0,4937 | 0,5286 |
| GP: Polinomial | 0,6496 | 0,5028 | 0,5352 | 0,4928 | 0,5301 |
| GP: PVK | 0,6157 | 0,4987 | 0,5287 | 0,5025 | 0,5385 |
| GP: RBF | 0,6083 | 0,5082 | 0,535 | 0,5012 | 0,5316 |
| MLP | 1,0134 | 0,4785 | 0,6675 | 0,4285 | 0,6488 |
| Redes RBF | 0,6402 | 0,508 | 0,5353 | 0,4964 | 0,5287 |
| SVR: Polinomial normal. | 0,997 | 0,4549 | 0,6728 | 0,4108 | 0,6392 |
| SVR: Polinomial | 0,998 | 0,4548 | 0,6727 | 0,411 | 0,6394 |
| SVR: PVK | 0,9967 | 0,4549 | 0,6729 | 0,4109 | 0,6392 |
| SVR: RBF | 0,9978 | 0,4548 | 0,673 | 0,4108 | 0,6394 |
| KNN K = 1 | 0,5926 | 0,4919 | 0,5271 | 0,5132 | 0,5499 |
| KNN K = 2 | 0,5926 | 0,5005 | 0,5302 | 0,5034 | 0,5353 |
| KNN K = 3 | 0,5926 | 0,5091 | 0,5333 | 0,5034 | 0,5353 |
| KNN K = 4 | 0,5926 | 0,5091 | 0,5333 | 0,5034 | 0,5353 |
| KNN K = 5 | 0,5926 | 0,5091 | 0,5333 | 0,5034 | 0,5353 |

La Tabla 6.14 referencia a la categorización *OpinionAporteIrrelevante* y a los resultados logrados por los diferentes métodos de predicción para el usuario *alfonsoblog*. El clasificador SVR con núcleo normalizado ofrece los mejores valores, tanto en la fase de entrenamiento como en la fase de prueba, respecto a las tasas de error MAE y RMSE. En la fase de entrenamiento, el SVR con núcleo normalizado devuelve un valor en la tasa de error MAE entorno al 45 %, mientras que éste mismo clasificador logra un valor en la tasa de error RMSE cercano al 67 %. Por otra parte, en la fase de prueba, dicho clasificador obtiene como valores 0,4108 y 0,6392, en cuanto a las tasas de error MAE y RMSE se refiere. Finalmente, el valor de salida del mejor modelo predictivo (SVR con núcleo normalizado) para el usuario *alfonsoblog* es 0,997. Lo cual indica que el próximo comentario a realizar por este usuario será de tipo Irrelevante.

Teniendo en cuenta los resultados devueltos por el resto de algoritmos predictivos para el usuario *alfonsoblog*, todos ellos ofrecen valores en torno a 1 (destacando los distintos clasificadores SVR con sus diferentes configuraciones de núcleo, así como el algoritmo MLP). Es decir, todos los modelos de predicción pronostican que el siguiente comentario a publicar por este usuario será de tipo Irrelevante.

**Tabla 6.15:** Resultados obtenidos en base a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, sobre la categorización *OpinionAporteIrrelevante* para el usuario *arolasecas*.

| | | Entrenamiento (N=101) | | Prueba (N=44) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,6166 | 0,464 | 0,4834 | 0,5036 | 0,5434 |
| GP: Polinomial | 0,6148 | 0,4647 | 0,4834 | 0,5027 | 0,5429 |
| GP: PVK | 0,6045 | 0,4661 | 0,4828 | 0,5089 | 0,5469 |
| GP: RBF | 0,6086 | 0,4762 | 0,4871 | 0,4986 | 0,5275 |
| MLP | 0,3657 | 0,5256 | 0,5465 | 0,5665 | 0,6065 |
| Redes RBF | 0,5597 | 0,4691 | 0,485 | 0,5013 | 0,5409 |
| SVR: Polinomial normal. | 0,9994 | 0,4006 | 0,5905 | 0,4387 | 0,6113 |
| SVR: Polinomial | 0,998 | 0,4061 | 0,6359 | 0,3869 | 0,6207 |
| SVR: PVK | 0,9997 | 0,3961 | 0,629 | 0,4876 | 0,6918 |
| SVR: RBF | 0,9991 | 0,406 | 0,6364 | 0,3867 | 0,6211 |
| KNN K = 1 | 0,6 | 0,4661 | 0,4828 | 0,5078 | 0,5462 |
| KNN K = 2 | 0,6 | 0,4661 | 0,4828 | 0,5078 | 0,5462 |
| KNN K = 3 | 0,6 | 0,4661 | 0,4828 | 0,5078 | 0,5462 |
| KNN K = 4 | 0,6 | 0,4661 | 0,4828 | 0,5078 | 0,5462 |
| KNN K = 5 | 0,6 | 0,4661 | 0,4828 | 0,5078 | 0,5462 |

La Tabla 6.15 presenta los valores ofrecidos por la tasa del error absoluto medio, por la tasa de la raíz del error cuadrático medio (tanto para la fase de entrenamiento como para la fase de prueba), así como los resultados finales sobre qué tipo de comentario realizará el usuario *arolasecas* en su próxima publicación. En este caso, el modelo de predicción que mejores resultados logra es el SVR con un núcleo basado en funciones de base radial. Dicho algoritmo obtiene 0,406 de valor de tasa de error MAE y 0,6364 en la tasa de error RMSE, ambos durante la fase de entrenamiento. En cuanto a la fase de prueba, este clasificador logra valores de 0,3867 y 0,6211 para las tasas de error MAE y RMSE, respectivamente. Con todo ello, el usuario *arolasecas* publicará un comentario de tipo Irrelevante, ya que el resultado ofrecido por el mejor clasificador (SVR con núcleo RBF) es de 0,9991.

Cabe destacar el comportamiento de los clasificadores $K$-vecinos más cercanos regresivos; debido a que, tanto en la fase de entrenamiento como en la fase de prueba, los valores devueltos por estos algoritmos en todas sus variantes de $K$ son iguales. Por lo cual, también los resultados ofrecidos como salida por el modelo de predicción son iguales. En esta clasificación también existe homogeneidad en los modelos de predicción (excepto el clasificador MLP) en cuanto a los resultados ofrecidos por los clasificadores: el próximo comentario publicado por el usuario *arolasecas* será de tipo Irrelevante.

### 6.4.2 Serie temporal *NormalPolemicoMuyPolemico*

A continuación, esta sección muestra los resultados logrados gracias a la utilización de los diferentes modelos predictivos de series temporales para la clasificación referente al tipo de comentario: (i) Normal, (ii) Polémico o (iii) Muy Polémico. Dicha sección expone los valores obtenidos tanto por el error absoluto medio como por la raíz del error cuadrático medio, así como el resultado de la predicción. Ambas tasas de error se muestran para ambas fases: (i) entrenamiento y (ii) prueba.

**Tabla 6.16:** Resultados logrados en cuanto a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, respecto a la clasificación *NormalPolemicoMuyPolemico* para el usuario *gabi10*.

| | | **Entrenamiento (N=143)** | | **Prueba (N=62)** | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,4621 | 0,5192 | 0,5499 | 0,4913 | 0,5105 |
| GP: Polinomial | 0,4621 | 0,5209 | 0,5507 | 0,4991 | 0,5165 |
| GP: PVK | 0,4346 | 0,5094 | 0,5438 | 0,4903 | 0,5119 |
| GP: RBF | 0,4577 | 0,5227 | 0,5517 | 0,4959 | 0,5144 |
| MLP | 0,3313 | 0,5088 | 0,5613 | 0,4418 | 0,4859 |
| Redes RBF | 0,4776 | 0,5121 | 0,5467 | 0,496 | 0,517 |
| SVR: Polinomial normal. | 0,0008 | 0,4669 | 0,6968 | 0,2609 | 0,5341 |
| SVR: Polinomial | 0,004 | 0,4689 | 0,7222 | 0,2597 | 0,537 |
| SVR: PVK | 0,0001 | 0,4478 | 0,69 | 0,2849 | 0,5483 |
| SVR: RBF | 0,0012 | 0,4686 | 0,7231 | 0,2588 | 0,5381 |
| KNN K = 1 | 0,44 | 0,5023 | 0,5421 | 0,4971 | 0,5234 |
| KNN K = 2 | 0,44 | 0,5054 | 0,5433 | 0,4901 | 0,5127 |
| KNN K = 3 | 0,44 | 0,5179 | 0,5485 | 0,4964 | 0,5152 |
| KNN K = 4 | 0,44 | 0,519 | 0,549 | 0,4964 | 0,5152 |
| KNN K = 5 | 0,44 | 0,519 | 0,549 | 0,4964 | 0,5152 |

La Tabla 6.16 ofrece los resultados logrados tanto para la tasa de error MAE como para la tasa de error RMSE, sobre la categorización *NormalPolemicoMuyPolemico* del usuario *gabi10*. En la fase de entrenamiento, los resultados obtenidos para ambas tasas (MAE y RMSE) son 0,4686 y 0,7231, respectivamente. Mientras que los resultados obtenidos para estas mismas tasas de error, en este caso en la fase de prueba, son 0,25 (en la tasa MAE) y 0,5381 (en la tasa RMSE). Ambos valores en las tasas de error han sido obtenidos por el clasificador SVR con un núcleo basado en funciones de base radial, el cual a su vez devuelve como resultado predictivo 0,0012; es decir, en este caso el algoritmo estima que el próximo comentario a realizar por el usuario *gabi10* será un comentario de tipo Normal.

**Tabla 6.17:** Resultados obtenidos en base a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, sobre la categorización *NormalPolemicoMuyPolemico* para el usuario *pavlenka*.

| | | Entrenamiento (N=151) | | Prueba (N=65) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,5163 | 0,6117 | 0,6963 | 0,5999 | 0,6779 |
| GP: Polinomial | 0,5163 | 0,6194 | 0,6981 | 0,609 | 0,6816 |
| GP: PVK | 0,6065 | 0,5905 | 0,6762 | 0,6041 | 0,7048 |
| GP: RBF | 0,5216 | 0,6221 | 0,6983 | 0,6082 | 0,6802 |
| MLP | 0,6728 | 0,648 | 0,7114 | 0,6082 | 0,6764 |
| Redes RBF | 0,5264 | 0,6219 | 0,6985 | 0,6098 | 0,6822 |
| SVR: Polinomial normal. | 0,0019 | 0,5167 | 0,8682 | 0,6152 | 0,9107 |
| SVR: Polinomial | 0,0036 | 0,5171 | 0,8673 | 0,6155 | 0,91 |
| SVR: PVK | 0,0014 | 0,5037 | 0,837 | 0,6153 | 0,9104 |
| SVR: RBF | 0,0025 | 0,5169 | 0,8676 | 0,6153 | 0,9101 |
| KNN K = 1 | 0,6111 | 0,5774 | 0,6732 | 0,6057 | 0,7307 |
| KNN K = 2 | 0,6111 | 0,5774 | 0,6732 | 0,6057 | 0,7307 |
| KNN K = 3 | 0,6111 | 0,5774 | 0,6732 | 0,6057 | 0,7307 |
| KNN K = 4 | 0,6111 | 0,5818 | 0,6797 | 0,6159 | 0,7354 |
| KNN K = 5 | 0,611 | 0,5983 | 0,6872 | 0,5873 | 0,6891 |

La Tabla 6.17 muestra todos los valores obtenidos por los distintos clasificadores de predicción, en relación a la clasificación *NormalPolemicoMuyPolemico* referente al usuario *pavlenka*. De todos ellos, el algoritmo que mejor rendimiento ofrece a la predicción es el $K$-vecinos más cercanos regresivos. En la fase de entrenamiento, tanto la tasa de error MAE como la tasa de error RMSE, ofrecen resultados altos en sus porcentajes de error: 59,83 % y 68,72 %, respectivamente. Por otra parte, en la fase de prueba, también dichas tasas logran porcentajes de error elevados: 58,73 % en la tasa de error MAE y 68,91 % en la tasa de error RMSE. Aunque la tasa de error de MAE es elevada, el resultado obtenido por la tasa de error de RMSE es notable; ya que a menor diferencia entre ambos valores (MAE y RMSE), menor será la variación en los errores individuales de la muestra.

En referencia al tipo de comentario a redactar por el usuario *pavlenka* en su próxima publicación, el algoritmo $K$-vecinos más cercanos regresivos con un valor de $K = 5$ devuelve una salida predictiva de 0,611. En este caso, dicho usuario publicará un comentario de tipo Polémico. Este resultado es apoyado por el resto de clasificadores de predicción (excepto por los algoritmos SVR con sus respectivos núcleos) obteniendo todos ellos valores cercanos a 1 o lo que es lo mismo, un comentario de tipo Polémico.

**Tabla 6.18:** Resultados logrados en cuanto a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, respecto a la clasificación *NormalPolemicoMuyPolemico* para el usuario *batis*.

| | | Entrenamiento (N=261) | | Prueba (N=112) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,3502 | 0,5209 | 0,5934 | 0,4871 | 0,5466 |
| GP: Polinomial | 0,3868 | 0,5266 | 0,5993 | 0,4854 | 0,5413 |
| GP: PVK | 0,3914 | 0,4783 | 0,5802 | 0,453 | 0,5498 |
| GP: RBF | 0,366 | 0,5106 | 0,601 | 0,4593 | 0,5261 |
| MLP | 0,4497 | 0,553 | 0,6114 | 0,5191 | 0,5595 |
| Redes RBF | 0,3695 | 0,4985 | 0,5962 | 0,4512 | 0,5349 |
| SVR: Polinomial normal. | 0 | 0,3755 | 0,716 | 0,268 | 0,5822 |
| SVR: Polinomial | 0,0025 | 0,3767 | 0,7153 | 0,2695 | 0,5811 |
| SVR: PVK | 0,0032 | 0,3604 | 0,6626 | 0,3128 | 0,6191 |
| SVR: RBF | 0,0018 | 0,3761 | 0,7156 | 0,2688 | 0,5817 |
| KNN K = 1 | 0,3939 | 0,4768 | 0,5799 | 0,4548 | 0,5534 |
| KNN K = 2 | 0,3939 | 0,4768 | 0,5799 | 0,4548 | 0,5534 |
| KNN K = 3 | 0,3939 | 0,4768 | 0,5799 | 0,4548 | 0,5534 |
| KNN K = 4 | 0,3939 | 0,4774 | 0,5802 | 0,4548 | 0,5534 |
| KNN K = 5 | 0,3939 | 0,4774 | 0,5802 | 0,4548 | 0,5534 |

La Tabla 6.18 presenta al usuario *batis* y a su categorización *NormalPolemicoMuyPolemico*. En la fase de entrenamiento, el clasificador SVR con núcleo normalizado logra los mejores resultados en las tasas de error absoluto medio y en la raíz del error cuadrático medio: 0,3755 y 0,716, respectivamente. Al igual que en la fase de entrenamiento, en la fase de prueba el algoritmo SVR con núcleo normalizado obtiene el valor 0,268 en la tasa de error MAE y el valor 0,5822 en la tasa de error RMSE. Debido a estos valores en ambas tasas de error, el resultado en la predicción para la clasificación *NormalPolemicoMuyPolemico* del usuario *batis* es 0; es decir, el usuario en cuestión realizará un comentario de tipo Normal la próxima vez que publique un comentario en el sitio web social de noticias *Menéame*.

En cuanto al resto de clasificadores y el comportamiento que ofrecen, destacan los $K$-vecinos más cercanos. Estos algoritmos, en todas sus variantes de $K$, obtienen los mismos valores en la fase de prueba, tanto para la tasa de error MAE (0,4548) como para la tasa de error RMSE (0,5534). Por ello, todas las salidas predictivas de los $K$-vecinos más cercanos regresivos devuelven el mismo valor: 0,3939. Con lo cual, su rendimiento está en la misma línea que el mejor de los clasificadores (SVR con núcleo normalizado) al estimar como tipo Normal la siguiente publicación a realizar por el usuario *batis*.

**Tabla 6.19:** Resultados obtenidos en base a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, sobre la categorización *NormalPolemicoMuyPolemico* para el usuario *hangla*.

| | | Entrenamiento (N=242) | | Prueba (N=105) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,3279 | 0,4771 | 0,5731 | 0,4111 | 0,462 |
| GP: Polinomial | 0,3279 | 0,4845 | 0,5721 | 0,4179 | 0,4624 |
| GP: PVK | 0,2912 | 0,459 | 0,5619 | 0,3936 | 0,4646 |
| GP: RBF | 0,3087 | 0,4728 | 0,5739 | 0,4044 | 0,46 |
| MLP | 0,252 | 0,4475 | 0,5663 | 0,3687 | 0,4504 |
| Redes RBF | 0,2819 | 0,4691 | 0,5728 | 0,3929 | 0,4568 |
| SVR: Polinomial normal. | 0,0011 | 0,3227 | 0,6614 | 0,2101 | 0,4972 |
| SVR: Polinomial | 0,0016 | 0,323 | 0,6609 | 0,2105 | 0,4968 |
| SVR: PVK | 0,0017 | 0,3181 | 0,6434 | 0,2153 | 0,5027 |
| SVR: RBF | 0,0006 | 0,3227 | 0,6614 | 0,21 | 0,4973 |
| KNN K = 1 | 0,2889 | 0,4518 | 0,5588 | 0,4078 | 0,5023 |
| KNN K = 2 | 0,2889 | 0,4576 | 0,566 | 0,3887 | 0,4579 |
| KNN K = 3 | 0,2889 | 0,4576 | 0,566 | 0,3887 | 0,4579 |
| KNN K = 4 | 0,2889 | 0,4576 | 0,566 | 0,3887 | 0,4579 |
| KNN K = 5 | 0,2889 | 0,4579 | 0,566 | 0,3881 | 0,457 |

Respecto a la categorización *NormalPolemicoMuyPolemico* sobre el usuario *hangla*, la Tabla 6.19 expone los resultados obtenidos para las tasas de error MAE y RMSE, así como el resultado logrado por todos los clasificadores seleccionados para tal tarea, tanto en la fase de entrenamiento como en la fase de prueba. En este caso, el algoritmo SVR con núcleo basado en funciones de base radial obtiene los mejores resultados en la fase de entrenamiento: 32,27 % en la tasa de error MAE y 66,14 % en la tasa de error RMSE. En la misma línea, dicho clasificador también logra los mejores valores en la fase de prueba para ambas tasas de error MAE y RMSE: 21 % y 49,73 %, al respecto. En cuanto a la salida devuelta por dicho modelo de predicción, su valor es de 0,0006. Esto conlleva a deducir que el siguiente comentario será de tipo Normal, en el momento en el que el usuario *hangla* publique un contenido.

Cabe destacar el comportamiento de todos los algoritmos SVR con sus respectivos núcleos, ya que ofrecen todos ellos resultados de tasas de error bajas (tanto en la fase de entrenamiento como en la fase de prueba) y homogeneidad en las salidas predictivas, con valores cercanos a 0. Valor el cual es ratificado por el resto de modelos de predicción automática al lograr resultados inferiores a 0,3279. Es decir, todos los clasificadores predicen una próxima publicación de tipo Normal realizada por el usuario *hangla*.

**Tabla 6.20:** Resultados logrados en cuanto a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, respecto a la clasificación *NormalPolemicoMuyPolemico* para el usuario *willies*.

| | | Entrenamiento (N=140) | | Prueba (N=61) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,5579 | 0,5805 | 0,6439 | 0,5299 | 0,5589 |
| GP: Polinomial | 0,5734 | 0,5791 | 0,6437 | 0,5246 | 0,5527 |
| GP: PVK | 0,5155 | 0,5477 | 0,6331 | 0,4945 | 0,5375 |
| GP: RBF | 0,4875 | 0,571 | 0,6444 | 0,5075 | 0,5426 |
| MLP | 0,2726 | 0,4958 | 0,6936 | 0,3702 | 0,5336 |
| Redes RBF | 0,4235 | 0,5664 | 0,6431 | 0,4939 | 0,5276 |
| SVR: Polinomial normal. | 0,0032 | 0,4573 | 0,7918 | 0,2955 | 0,6 |
| SVR: Polinomial | 0,0035 | 0,4576 | 0,7916 | 0,2959 | 0,5997 |
| SVR: PVK | 0,0422 | 0,4294 | 0,7722 | 0,2963 | 0,5992 |
| SVR: RBF | 0,0053 | 0,4573 | 0,7915 | 0,2956 | 0,5999 |
| KNN K = 1 | 0,5 | 0,5388 | 0,6315 | 0,4942 | 0,5395 |
| KNN K = 2 | 0,5 | 0,5438 | 0,6343 | 0,4942 | 0,5395 |
| KNN K = 3 | 0,5 | 0,5438 | 0,6343 | 0,4942 | 0,5395 |
| KNN K = 4 | 0,5 | 0,5438 | 0,6343 | 0,4942 | 0,5395 |
| KNN K = 5 | 0,5 | 0,5554 | 0,638 | 0,4942 | 0,5395 |

La Tabla 6.20 hace referencia a los resultados obtenidos en la clasificación *NormalPolemicoMuyPolemico* para el usuario *willies*. Tanto en la fase de entrenamiento como en la fase de prueba, el modelo de predicción que mejor comportamiento ha tenido es el SVR con núcleo normalizado. En la fase de entrenamiento, dicho clasificador logra valores de 0,4573 para la tasa de error MAE y 0,7918 para la tasa de error RMSE, mientras que en la fase de prueba ofrece valores de 0,2955 para la tasa de error MAE y 0,6 para la tasa de error RMSE. Con todo ello, el SVR con núcleo normalizado devuelve como salida de predicción el valor 0,0032 o lo que es lo mismo, predice que el comentario sucesivo a redactar por el usuario *willies* será de tipo Normal.

Por otra parte, vuelve a destacar el rendimiento de los clasificadores $K$-vecinos más cercanos regresivos. En la fase de entrenamiento obtienen valores muy semejantes entre ellos con apenas diferencia numérica; no obstante, en la fase de prueba, los valores logrados por ambas tasas de error (MAE y RMSE) son los mismos para todas las configuraciones de $K$. Debido a estos resultados, todos los valores de las salidas de estos 5 modelos predictivos seleccionados anteriormente para tal tarea son totalmente iguales: 0,5.

**Tabla 6.21:** Resultados obtenidos en base a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, sobre la categorización *NormalPolemicoMuyPolemico* para el usuario *tachyon*.

| | | Entrenamiento (N=249) | | Prueba (N=108) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,2749 | 0,4184 | 0,4994 | 0,4204 | 0,5125 |
| GP: Polinomial | 0,2749 | 0,4251 | 0,5006 | 0,4234 | 0,5091 |
| GP: PVK | 0,217 | 0,3925 | 0,4936 | 0,3996 | 0,5126 |
| GP: RBF | 0,2588 | 0,4114 | 0,5028 | 0,4085 | 0,5121 |
| MLP | 0,0083 | 0,2838 | 0,5657 | 0,2649 | 0,562 |
| Redes RBF | 0,2206 | 0,3956 | 0,4962 | 0,4047 | 0,5164 |
| SVR: Polinomial normal. | 0 | 0,2762 | 0,5634 | 0,266 | 0,5709 |
| SVR: Polinomial | 0,0014 | 0,2779 | 0,5766 | 0,2601 | 0,5766 |
| SVR: PVK | 0,0031 | 0,2663 | 0,5658 | 0,2883 | 0,5965 |
| SVR: RBF | 0,0001 | 0,2773 | 0,577 | 0,2596 | 0,5771 |
| KNN K = 1 | 0,2185 | 0,3905 | 0,493 | 0,3986 | 0,5156 |
| KNN K = 2 | 0,2185 | 0,3926 | 0,4941 | 0,4082 | 0,5204 |
| KNN K = 3 | 0,2185 | 0,3947 | 0,4952 | 0,4034 | 0,5135 |
| KNN K = 4 | 0,2185 | 0,3947 | 0,4952 | 0,4034 | 0,5135 |
| KNN K = 5 | 0,2185 | 0,3947 | 0,4961 | 0,399 | 0,5102 |

La Tabla 6.21 referencia a los resultados del usuario *tachyon*, así como su categorización *NormalPolemicoMuyPolemico*. En esta clasificación del usuario *tachyon*, los valores obtenidos para ambas tasas de error en ambas fases son muy similares. El algoritmo que mejores valores ofrece es el SVR con núcleo basado en funciones de base radial. En la fase de entrenamiento, logra 0,2773 en la tasa de error MAE y 0,577 en la tasa de error RMSE. En este mismo sentido, en la fase de prueba, obtiene 0,2596 para la tasa de error MAE y 0,5771 para la tasa de error RMSE. Por ello, el resultado que devuelve dicho clasificador como salida es 0,0001. El usuario *tachyon*, en el momento en el que haga un comentario en el sitio web social de noticias *Menéame* será de tipo Normal.

En cuanto al rendimiento del resto de modelos de predicción, destaca que todos ellos obtienen valores en sus salidas predictivas muy similares entre sí, todas ellos cercanos a 0. Con lo cual, confirman el resultado devuelto por el mejor de los clasificadores para esta categorización *NormalPolemicoMuyPolemico* del usuario *tachyon* en referencia al tipo de comentario a realizar por dicho usuario: comentario de tipo Normal.

**Tabla 6.22:** Resultados logrados en cuanto a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, respecto a la clasificación *NormalPolemicoMuyPolemico* para el usuario *berthax*.

| | | Entrenamiento (N=156) | | Prueba (N=67) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,2278 | 0,3452 | 0,4454 | 0,3549 | 0,4603 |
| GP: Polinomial | 0,2278 | 0,339 | 0,4446 | 0,3499 | 0,462 |
| GP: PVK | 0,2601 | 0,3509 | 0,443 | 0,3641 | 0,4618 |
| GP: RBF | 0,233 | 0,3552 | 0,4461 | 0,3643 | 0,4601 |
| MLP | 0,4915 | 0,4933 | 0,5088 | 0,4944 | 0,514 |
| Redes RBF | 0,2476 | 0,354 | 0,4456 | 0,3648 | 0,4609 |
| SVR: Polinomial normal. | 0,0019 | 0,2251 | 0,4993 | 0,2396 | 0,5177 |
| SVR: Polinomial | 0,0033 | 0,2257 | 0,4989 | 0,2401 | 0,5173 |
| SVR: PVK | 0,002 | 0,2254 | 0,4991 | 0,2401 | 0,5175 |
| SVR: RBF | 0,0019 | 0,2251 | 0,4993 | 0,2696 | 0,5178 |
| KNN K = 1 | 0,2553 | 0,3469 | 0,4427 | 0,3616 | 0,4636 |
| KNN K = 2 | 0,2553 | 0,3469 | 0,4427 | 0,3616 | 0,4636 |
| KNN K = 3 | 0,2553 | 0,3518 | 0,4438 | 0,359 | 0,4583 |
| KNN K = 4 | 0,2553 | 0,3518 | 0,4438 | 0,359 | 0,4583 |
| KNN K = 5 | 0,2553 | 0,3518 | 0,4438 | 0,359 | 0,4583 |

La Tabla 6.22 presenta al usuario *berthax*, así como los valores obtenidos para la clasificación *NormalPolemicoMuyPolemico*, en base a la tasa de error absoluto medio y a la raíz del error cuadrático medio. En la fase de entrenamiento, los valores logrados por ambas tasas de error son porcentualmente bajos comparados con otros casos: la tasa de error MAE devuelve como media de todos los clasificadores un valor de 32,58 %, mientras que la tasa de error RMSE devuelve como media de valores un resultado de 46,31 %. Para ambos casos, el mejor comportamiento de entre todos los modelos de predicción es el correspondiente al algoritmo SVR con núcleo normalizado: 22,51 % de tasa de error MAE y 49,93 % en la tasa de error RMSE.

Por su parte, en la fase de prueba, los resultados ofrecidos son muy parecidos a la fase de entrenamiento. La media de los 15 clasificadores es de 33,88 % en base a la tasa de error MAE y de 47,94 % para la tasa de error RMSE, siendo en este caso el clasificador SVR con núcleo normalizado el mejor de ellos: 23,96 % respecto a la tasa de error MAE y 51,77 % en la tasa de error RMSE.

Una vez dispuestos ambos valores, el clasificador en cuestión ofrece una salida predictiva con un valor 0,0019; o lo que es lo mismo, predice que el siguiente texto a redactar en forma de comentario por el usuario *berthax* en *Menéame* será de tipo Normal. Este valor es corroborado por el resto de modelos de predicción, ya que como media ofrecen un valor de 0,1982 en la salida predictiva.

**Tabla 6.23:** Resultados obtenidos en base a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, sobre la categorización *NormalPolemicoMuyPolemico* para el usuario *hsc*.

| | | Entrenamiento (N=219) | | Prueba (N=94) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,2353 | 0,3753 | 0,46 | 0,4021 | 0,4878 |
| GP: Polinomial | 0,2353 | 0,3696 | 0,4604 | 0,3982 | 0,4915 |
| GP: PVK | 0,2002 | 0,358 | 0,457 | 0,3873 | 0,4876 |
| GP: RBF | 0,2315 | 0,3643 | 0,4608 | 0,3965 | 0,4982 |
| MLP | 0,5702 | 0,5549 | 0,5714 | 0,5528 | 0,5753 |
| Redes RBF | 0,2326 | 0,3647 | 0,4612 | 0,3981 | 0,5005 |
| SVR: Polinomial normal. | 0,0015 | 0,2292 | 0,5138 | 0,2878 | 0,5732 |
| SVR: Polinomial | 0,0004 | 0,2285 | 0,5145 | 0,2874 | 0,5741 |
| SVR: PVK | 0,0031 | 0,2298 | 0,5136 | 0,2884 | 0,5732 |
| SVR: RBF | 0,0036 | 0,23 | 0,5134 | 0,2885 | 0,5729 |
| KNN K = 1 | 0,1957 | 0,3535 | 0,4567 | 0,3841 | 0,4919 |
| KNN K = 2 | 0,1957 | 0,3554 | 0,4572 | 0,3861 | 0,4882 |
| KNN K = 3 | 0,1957 | 0,3554 | 0,4572 | 0,3861 | 0,4882 |
| KNN K = 4 | 0,1957 | 0,3589 | 0,4582 | 0,3889 | 0,489 |
| KNN K = 5 | 0,1957 | 0,3589 | 0,4582 | 0,3889 | 0,489 |

La Tabla 6.23 muestra los valores de la categorización *NormalPolemicoMuyPolemico*, en referencia al usuario *hsc*. En la fase de entrenamiento, el clasificador SVR con núcleo polinomial normalizado arroja los mejores resultados tanto para la tasa de error MAE, como para la tasa de error RMSE: 0,2285 y 0,5145 respectivamente. En la misma línea, dicho clasificador logra los resultados más óptimos para esta clasificación: 0,2874 en la tasa de error MAE y 0,5741 en la tasa de error RMSE. Gracias a estos valores, el modelo predictivo SVR con núcleo polinomial normalizado ofrece una salida con un resultado de 0,0004, lo cual indica que el próximo comentario del usuario *hsc* será un comentario de tipo Normal.

Respecto al comportamiento del resto de modelos de predicción empleados, destacan por encima de todos los $K$-vecinos más cercanos regresivos. En la fase de entrenamiento y en la fase de prueba, todas las configuraciones del término $K$ utilizadas (de 1 a 5) para estos clasificadores logran valores muy similares tanto para la tasa de error absoluto medio como para la tasa de la raíz del error cuadrático medio, pero en el caso del valor de salida de dichos algoritmos ofrecen el mismo resultado: 0,1957.

**Tabla 6.24:** Resultados logrados en cuanto a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, respecto a la clasificación *NormalPolemicoMuyPolemico* para el usuario *psalbendea*.

| | | Entrenamiento (N=212) | | Prueba (N=92) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,3088 | 0,3728 | 0,4709 | 0,4812 | 0,6257 |
| GP: Polinomial | 0,2911 | 0,3721 | 0,4711 | 0,4794 | 0,6225 |
| GP: PVK | 0,2932 | 0,355 | 0,4646 | 0,4846 | 0,6386 |
| GP: RBF | 0,2471 | 0,3649 | 0,4712 | 0,4731 | 0,6264 |
| MLP | 0,6058 | 0,5832 | 0,6064 | 0,613 | 0,6587 |
| Redes RBF | 0,2329 | 0,3653 | 0,4716 | 0,4726 | 0,6274 |
| SVR: Polinomial normal. | 0,0004 | 0,2276 | 0,5222 | 0,3919 | 0,7213 |
| SVR: Polinomial | 0,002 | 0,227 | 0,5226 | 0,3916 | 0,7215 |
| SVR: PVK | 0,0033 | 0,2219 | 0,5183 | 0,4162 | 0,7257 |
| SVR: RBF | 0,0005 | 0,2282 | 0,5218 | 0,3926 | 0,7213 |
| KNN K = 1 | 0,2857 | 0,3477 | 0,4638 | 0,4906 | 0,6496 |
| KNN K = 2 | 0,2857 | 0,3477 | 0,4638 | 0,4906 | 0,6496 |
| KNN K = 3 | 0,2857 | 0,3477 | 0,4642 | 0,4861 | 0,6475 |
| KNN K = 4 | 0,2857 | 0,3512 | 0,4652 | 0,4867 | 0,645 |
| KNN K = 5 | 0,2857 | 0,3512 | 0,4652 | 0,4867 | 0,645 |

En referencia al usuario *psalbendea* y a sus correspondientes resultados para la clasificación *NormalPolemicoMuyPolemico*, la Tabla 6.24 expone dichos valores en términos de tasa de error MAE, tasa de erro RMSE y salida predictiva de los modelos. En la fase de entrenamiento, el algoritmo que mejor rendimiento obtiene al lograr un menor valor de tasa de error MAE es SVR con núcleo polinomial normalizado con un 22,7 %. En cuanto a la tasa de error RMSE, el mismo clasificador ofrece un resultado de 0,5226. A su vez, en la fase de prueba, el algoritmo SVR con núcleo polinomial normalizado logra un valor de 0,3916 en la tasa de error MAE y 0,7215 en la tasa de error RMSE. Con todo ello, dicho modelo predice que el siguiente comentario a publicar por el usuario *psalbendea* será de tipo Normal, ya que la salida predictiva de tal modelo devuelve un valor de 0,002.

En el caso del usuario en cuestión y para esta clasificación *NormalPolemicoMuyPolemico*, los clasificadores $K$-vecinos más cercanos regresivos vuelven a actuar al unísono en cuanto al valor de la salida de predicción se refiere. Todos ellos, desde $K = 1$ hasta $K = 5$, ofrecen el mismo resultado como predicción futura: 0,2857. Es decir, al igual que el algoritmo SVR de núcleo polinomial normalizado, los $K$-vecinos más cercanos regresivos optan por un comentario tipo Normal como próximo comentario a realizar po dicho usuario.

**Tabla 6.25:** Resultados obtenidos en base a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, sobre la categorización *NormalPolemicoMuyPolemico* para el usuario *amadorcea*.

| | | Entrenamiento (N=74) | | Prueba (N=32) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,5475 | 0,579 | 0,6753 | 0,5683 | 0,6465 |
| GP: Polinomial | 0,5476 | 0,5813 | 0,6704 | 0,5873 | 0,6623 |
| GP: PVK | 0,3721 | 0,5385 | 0,6508 | 0,5729 | 0,6848 |
| GP: RBF | 0,4326 | 0,5767 | 0,6762 | 0,5556 | 0,6484 |
| MLP | 0,6603 | 0,6601 | 0,7153 | 0,6971 | 0,7531 |
| Redes RBF | 0,4785 | 0,5549 | 0,666 | 0,5641 | 0,6666 |
| SVR: Polinomial normal. | 0,004 | 0,4326 | 0,8047 | 0,3443 | 0,7281 |
| SVR: Polinomial | 0,0032 | 0,4324 | 0,8044 | 0,3445 | 0,7283 |
| SVR: PVK | 0,0024 | 0,379 | 0,7505 | 0,3796 | 0,747 |
| SVR: RBF | 0,0017 | 0,4325 | 0,8048 | 0,3443 | 0,7285 |
| KNN K = 1 | 0,3333 | 0,5199 | 0,644 | 0,5526 | 0,7102 |
| KNN K = 2 | 0,3333 | 0,5374 | 0,6533 | 0,6276 | 0,7513 |
| KNN K = 3 | 0,3333 | 0,5374 | 0,6604 | 0,5909 | 0,7052 |
| KNN K = 4 | 0,3333 | 0,5333 | 0,6633 | 0,5909 | 0,7052 |
| KNN K = 5 | 0,3333 | 0,5387 | 0,665 | 0,5909 | 0,7052 |

La Tabla 6.25 ofrece los resultados referentes a la clasificación *NormalPolemicoMuyPolemico* para el usuario *amadorcea* perteneciente al sitio web social de noticias *Menéame*. En cuanto a la fase de entrenamiento, la tasa de error MAE logra como mejor resultado 0,4326, mientras que la tasa de error RMSE obtiene como valor 0,8047. Ambos resultados ofrecidos por el mejor algoritmo predictivo en esta clasificación: el SVR con núcleo normalizado. Siguiendo en la misma línea, en lo referente a la fase de prueba, dicho clasificador logra un valor de 0,3443 en la tasa de error MAE y 0,7281 en la tasa de error RMSE. Por ello, la salida predictiva del modelo en cuestión es de 0,004, con lo que la próxima publicación a realizar por el usuario *amadorcea* será de tipo Normal.

Cabe destacar de manera negativa el rendimiento del modelo de predicción perceptrón multicapa. Este algoritmo obtiene unos resultados para ambas tasas de error, tanto en la fase de entrenamiento como en la fase de prueba, muy elevados. A su vez, el algoritmo predictivo devuelve como salida un valor de 0,6603; es decir, pronostica que el siguiente comentario del usuario *amadorcea* será de tipo Polémico, lo cual se aleja de la predicción del mejor clasificador para esta categorización y usuario.

**Tabla 6.26:** Resultados logrados en cuanto a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, respecto a la clasificación *NormalPolemicoMuyPolemico* para el usuario *anadelagua*.

| | | Entrenamiento (N=360) | | Prueba (N=155) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,2688 | 0,3 | 0,3814 | 0,281 | 0,3537 |
| GP: Polinomial | 0,273 | 0,2997 | 0,3804 | 0,2813 | 0,3542 |
| GP: PVK | 0,2643 | 0,2776 | 0,3775 | 0,2586 | 0,351 |
| GP: RBF | 0,1952 | 0,2852 | 0,3824 | 0,2647 | 0,3501 |
| MLP | 0,1762 | 0,2449 | 0,3813 | 0,2243 | 0,3522 |
| Redes RBF | 0,277 | 0,2833 | 0,3818 | 0,2635 | 0,3527 |
| SVR: Polinomial normal. | 0,0018 | 0,1728 | 0,4212 | 0,1426 | 0,3764 |
| SVR: Polinomial | 0,0023 | 0,1746 | 0,4205 | 0,1444 | 0,3755 |
| SVR: PVK | 0,0019 | 0,1681 | 0,4143 | 0,1434 | 0,376 |
| SVR: RBF | 0,0006 | 0,1745 | 0,4205 | 0,1444 | 0,3755 |
| KNN K = 1 | 0,2667 | 0,2734 | 0,3762 | 0,2566 | 0,3507 |
| KNN K = 2 | 0,2667 | 0,2772 | 0,3796 | 0,2566 | 0,3507 |
| KNN K = 3 | 0,2667 | 0,2772 | 0,3796 | 0,2566 | 0,3507 |
| KNN K = 4 | 0,2667 | 0,2772 | 0,3796 | 0,2566 | 0,3507 |
| KNN K = 5 | 0,2667 | 0,2772 | 0,3796 | 0,2566 | 0,3507 |

La Tabla 6.26 hace referencia al usuario *anadelagua* sobre la categorización *NormalPolemicoMuyPolemico*. En la fase de entrenamiento destacan todos los clasificadores predictivos tanto en los valores obtenidos para la tasa de erro MAE como para la tasa de error RMSE, todos ellos logran como media 25,08 % en la tasa de error MAE y 39,03 % en la tasa de error RMSE. Debido a ello, el comportamiento del modelo SVR con núcleo normalizado (el mejor algoritmo de predicción) ofrece uno de los mejores resultados teniendo en cuenta los distintos usuarios seleccionados y sus clasificaciones: 17,28 % en la tasa de error MAE y 42,12 % en la tasa de error RMSE.

Por su parte, en la fase de prueba, ocurre un hecho similar en consonancia con la fase de entrenamiento. Todos los algoritmos de predicción seleccionados logran como media en la tasa de error MAE un valor de 22,87 % y como media en la tasa de error RMSE un valor de 35,80 %. Por lo tanto, el clasificador SVR con núcleo normalizado ofrece los siguientes valores: 14,26 % respecto a la tasa de error MAE y 37,64 % en la tasa de error RMSE.

Con todo ello, el modelo de predicción SVR con núcleo normalizado, seleccionado debido a su gran rendimiento, devuelve como salida predictiva un valor de 0,0018, justificando de esta manera que el próximo comentario a escribir por el usuario *anadelagua* en el sitio web social de noticias *Menéame* será de tipo Normal.

**Tabla 6.27:** Resultados obtenidos en base a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, sobre la categorización *NormalPolemicoMuyPolemico* para el usuario *alfonsoblog*.

| | | Entrenamiento (N=88) | | Prueba (N=39) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,1 | 0,1861 | 0,3661 | 0,1653 | 0,2685 |
| GP: Polinomial | 0,1 | 0,1898 | 0,3692 | 0,1631 | 0,2639 |
| GP: PVK | 0,0909 | 0,1781 | 0,3633 | 0,1601 | 0,2642 |
| GP: RBF | 0,0994 | 0,1862 | 0,3703 | 0,1612 | 0,2671 |
| MLP | -0,0295 | 0,1331 | 0,3892 | 0,1058 | 0,2823 |
| Redes RBF | 0,0963 | 0,186 | 0,3706 | 0,1596 | 0,2655 |
| SVR: Polinomial normal. | 0,0015 | 0,1034 | 0,384 | 0,0782 | 0,2772 |
| SVR: Polinomial | 0,0032 | 0,1048 | 0,3837 | 0,0796 | 0,2768 |
| SVR: PVK | 0,0037 | 0,1051 | 0,3835 | 0,08 | 0,2768 |
| SVR: RBF | 0,0031 | 0,1047 | 0,3836 | 0,0795 | 0,2767 |
| KNN K = 1 | 0,0921 | 0,1751 | 0,3618 | 0,1653 | 0,2731 |
| KNN K = 2 | 0,0921 | 0,1751 | 0,3618 | 0,1653 | 0,2731 |
| KNN K = 3 | 0,0921 | 0,1797 | 0,365 | 0,1589 | 0,2642 |
| KNN K = 4 | 0,0921 | 0,1804 | 0,3678 | 0,1589 | 0,268 |
| KNN K = 5 | 0,0921 | 0,1804 | 0,3678 | 0,1589 | 0,268 |

La Tabla 6.27 presenta los resultados obtenidos respecto a la clasificación *NormalPolemicoMuyPolemico* por el usuario *alfonsoblog* en el sitio web social de noticias *Menéame*. En la fase de entrenamiento, los valores ofrecidos son óptimos, lo cual indican las tasas de error absoluto medio y la raíz del error cuadrático medio. En referencia a la tasa de error MAE, el mejor resultado logrado tiene un valor de 10,34 %; mientras que en referencia a la tasa de error RMSE, el mejor resultado es de 38,4 %. Por su parte, en la fase de prueba, la tasa de error MAE obtiene un valor de 7,82 % y la tasa de error RMSE alcanza un valor de 27,72 %, ambos siendo los mejores valores del conjunto. Gracias a estos óptimos resultados de predicción, el modelo SVR con núcleo normalizado es el clasificador que mejores valores devuelve para el usuario *alfonsoblog* en la categorización *NormalPolemicoMuyPolemico*.

Con todo ello, siendo el algoritmo SVR con núcleo normalizado uno de los modelos predictivos que mejor rendimiento aporta a la predicción de series temporales, la salida predictiva de dicho clasificador es 0,0015. Debido a este valor, y a los de ambas tasas de error en ambas fases, el próximo comentario a realizar por el usuario en cuestión será de tipo Normal. Cabe destacar que en este caso todos los modelos de predicción seleccionados para tal tarea corroboran este valor, devolviendo el algoritmo cuyo valor más difiere del SVR con núcleo normalizado un resultado de 0,1.

**Tabla 6.28:** Resultados logrados en cuanto a las tasas de error MAE y RMSE, en las fases de entrenamiento y prueba, respecto a la clasificación *NormalPolemicoMuyPolemico* para el usuario *arolasecas*.

| | | Entrenamiento (N=101) | | Prueba (N=44) | |
|---|---|---|---|---|---|
| **Clasificador** | **Resultado** | **MAE** | **RMSE** | **MAE** | **RMSE** |
| GP: Polinomial normal. | 0,2427 | 0,3931 | 0,5342 | 0,4977 | 0,6487 |
| GP: Polinomial | 0,2427 | 0,3947 | 0,5327 | 0,5005 | 0,6461 |
| GP: PVK | 0,2412 | 0,3819 | 0,5228 | 0,5052 | 0,6559 |
| GP: RBF | 0,2419 | 0,3938 | 0,5338 | 0,4983 | 0,6461 |
| MLP | 0,0223 | 0,2757 | 0,5851 | 0,459 | 0,7485 |
| Redes RBF | 0,2302 | 0,3935 | 0,5336 | 0,5021 | 0,6455 |
| SVR: Polinomial normal. | 0,0037 | 0,2494 | 0,5874 | 0,4325 | 0,7522 |
| SVR: Polinomial | 0,0035 | 0,2492 | 0,5875 | 0,4324 | 0,7524 |
| SVR: PVK | 0,0019 | 0,2485 | 0,5879 | 0,4323 | 0,753 |
| SVR: RBF | 0,0031 | 0,249 | 0,5877 | 0,4323 | 0,7526 |
| KNN K = 1 | 0,2462 | 0,3797 | 0,5218 | 0,5047 | 0,6576 |
| KNN K = 2 | 0,2462 | 0,381 | 0,522 | 0,5076 | 0,6579 |
| KNN K = 3 | 0,2462 | 0,381 | 0,522 | 0,5076 | 0,6579 |
| KNN K = 4 | 0,2462 | 0,3799 | 0,5223 | 0,5101 | 0,6679 |
| KNN K = 5 | 0,2462 | 0,3892 | 0,5244 | 0,5208 | 0,6697 |

La Tabla 6.28 referencia a los resultados obtenidos por los diferentes clasificadores seleccionados anteriormente, en base a las tasas de error MAE y RMSE por el usuario *arolasecas* dentro de la categorización *NormalPolemicoMuyPolemico*. En este caso, el algoritmo SVR con un núcleo basado en funciones de base radial es el mejor de todos los modelos de predicción. En la fase de entrenamiento, dicho clasificador ofrece un valor de 0,249 para la tasa de error MAE y 0,5877 para la tasa de error RMSE. En esta misma línea, el algoritmo en cuestión logra unos valores de 0,4323 y 0,7526 para la tasa de error MAE y RMSE, respectivamente. Por ello, la salida de predicción que este modelo SVR devuelve es de 0,0031. Este valor es el correspondiente a la publicación por parte del usuario *arolasecas* de un comentario de tipo Normal, en el sitio web social de noticias *Menéame*.

Respecto al resto de modelos de predicción, cabe destacar tanto los procesos *gaussianos* (con sus respectivos núcleos) como los $K$-vecinos más cercanos regresivos (con todas sus configuraciones). Ambos modelos de predicción obtiene valores muy similares tanto en la fase de entrenamiento como en la fase de prueba y para ambas tasas de error (MAE y RMSE). En cuanto a la salida predictiva que estos modelos devuelven, también dichos valores son muy similares entre sí.

En resumen, los valores obtenidos en las diversas tablas, en referencia a los 13 usuarios seleccionados de *Menéame*, ratifican cómo el empleo de las series temporales y su predicción puede ejercer de complemento o ayuda en la detección y prevención de usuarios trol en sitios web sociales.

Gracias a los resultados evaluados mediante las tasas de error MAE y RMSE, la categorización *OpinionAporteIrrelevante* logra como media de predicción un valor de 0,4559 evaluado mediante la tasa de error MAE, entre todos los usuarios experimentados. No obstante, dicha categorización obtiene un valor mínimo de 35,77 %, en la fase de entrenamiento para la tasa de error MAE. En referencia a la tasa de error RMSE, la media de todos los resultados obtenidos para tal tasa devuelve un valor de 0,6308, para la fase de entrenamiento, pero con un resultado mínimo de 49,83 %.

Por su parte, en la fase de prueba, el valor medio correspondiente a la tasa de error MAE disminuye hasta un 0,4485, en comparación con la fase de entrenamiento, ofreciendo un valor mínimo de 35,22 % de entre todos los usuarios. A su vez, la tasa de error RMSE logra un valor de 0,6239, teniendo como valor mínimo 50,13 % en toda la serie.

Dentro de la propia clasificación, destaca el modelo de predicción soporte vectorial regresivo, debido a que tanto en su versión de un núcleo normalizado como de un núcleo basado en funciones de base radial, son los algoritmos que mejores resultados devuelven para dicha serie temporal.

En cuanto a la clasificación *NormalPolemicoMuyPolemico*, ésta ofrece valores notables sobre la predicción de series temporales. Esta categorización es la más importante de ambas, ya que gracias a ella podremos debatir acerca de la idoneidad de restringir o no el acceso a los usuarios al sitio web social de noticias. Es decir, la clasificación *NormalPolemicoMuyPolemico* deberá establecer los rangos de actuación en la detección de usuarios trol, apoyada con la categorización *OpinionAporteIrrelevante*, debido a que esta categorización pronosticará el grado de polemicidad con el que el usuario publique su siguiente comentario.

En la fase de entrenamiento, la serie *NormalPolemicoMuyPolemico* logra de media un valor de 31,18 % para la tasa de error MAE y 60,69 % para la tasa de error RMSE. En este caso, el valor más bajo de toda la serie es 10,34 % en cuanto a la tasa MAE y 38,4 % para la tasa de error RMSE. Respecto a la fase de prueba, los resultados ofrecidos por esta clasificación también son óptimos. En referencia a la media de valores, la tasa de error MAE logra un 29,19 % y la tasa de error RMSE un 57,16 %. Ambas, a su vez, ofrecen valores mínimos sobresalientes: 7,82 % en la tasa de error MAE y 27,72 % en la tasa de error RMSE.

Respecto al comportamiento de los clasificadores predictivos empleados, cabe destacar la aportación del modelo de predicción SVR con núcleo normalizado, así como el SVR con núcleo basado en funciones de base radial. Ambas configuraciones del modelo logran resultados sobresalientes para la clasificación *NormalPolemicoMuyPolemico*, tanto en la fase de entrenamiento como en la fase de prueba y para los 13 usuarios seleccionados con anterioridad. Como nota negativa, en cuanto al rendimiento se refiere, el perceptrón multicapa se erige como el algoritmo que resultados más altos ofrece en base a las tasas de error MAE y RMSE.

## 6.5 **Sumario**

El capítulo 6 presenta el concepto de series temporales y su predicción aplicado al mundo de la informática, pero más concretamente a los sitios web sociales de noticias y a la detección de usuarios trol, con el fin de poder predecir la siguiente acción de un usuario en estos sitios: su próximo comentario. El poder determinar el siguiente comentario a publicar por cualquier usuario puede evitar diversos problemas en el sitio web social, como por ejemplo: (i) enfrentamientos verbales, (ii) difusión de *spam* o (iii) la aparición de usuarios trol y grupos de presión.

Para ello, en un primer lugar hemos realizado una breve introducción al mundo de las series temporales y su predicción, ordenando cronológicamente su aparición. Más adelante, hemos definido tales conceptos: las series temporales y la predicción de las series temporales. También, hemos expuesto nuestras principales contribuciones al mundo científico. Seguidamente, hemos explicado la descripción del método llevado a cabo para el análisis y predicción de las series temporales, mostrando los 13 usuarios seleccionados para tal tarea, así como sus características. A continuación, hemos explicado las características extraídas de los usuarios y sus comentarios con el fin de definir los conjuntos de datos que nos permitan crear las series temporales.

En segundo lugar, hemos descrito los modelos de predicción empleados en este capítulo, así como todas sus diferentes configuraciones: (i) el perceptrón multicapa, (ii) las redes neuronales basadas en funciones de base radial, (iii) los $K$-vecinos más cercanos regresivos, (iv) el soporte vectorial regresivo y (v) los procesos *gaussianos*.

En tercer lugar, hemos desarrollado la validación empírica, creando para ello 2 nuevos atributos, los cuales en conjunto con los anteriormente descritos

permiten la creación de las 2 series temporales a analizar: (i) *OpinionAporteIrrelevante* y (ii) *NormalPolemicoMuyPolemico*. Una vez creadas, hemos mostrado la metodología a seguir para esta validación empírica así como su evaluación mediante 2 tasas de error: (i) la tasa de error absoluto medio (tasa de error MAE) y (ii) la tasa de la raíz del error cuadrático medio (tasa de error RMSE).

Por último, hemos procedido a mostrar, y a su vez también discutir, todos los valores y resultados obtenidos para ambas categorizaciones (*OpinionAporteIrrelevante* y *NormalPolemicoMuyPolemico*), en base a las tasas de error MAE y RMSE, tanto para la fase de entrenamiento como para la fase de prueba.

*«El éxito es aprender a ir de fracaso en fracaso sin desesperarse.»*

Winston Leonard Spencer Churchill
(1874–1965)

CAPÍTULO

# 7

# Conclusiones

ESTE capítulo es el encargado de poner fin a esta tesis doctoral realizando un recorrido tanto por la hipótesis fundamental como por todos los objetivos marcados con anterioridad, con el fin de evaluar su cumplimiento gracias a las contribuciones descritas y realizadas a lo largo de esta investigación.

El resto del capítulo está estructurado de la siguiente manera. La sección 7.1 describe las principales contribuciones realizadas a lo largo de esta tesis, a la vez que corrobora la hipótesis fundamental. La sección 7.2 hace referencia a diversas limitaciones encontradas en el transcurso de la investigación. La sección 7.3 desarrolla el trabajo y líneas de investigación futuras del presente trabajo. Por último, la sección 7.4 expone las consideraciones finales de esta tesis doctoral.

## 7.1 Principales contribuciones

En primer lugar, presentamos la hipótesis fundamental de partida descrita anteriormente, con el fin de obtener su corroboración mediante el cumplimiento de todos los objetivos planteados:

> «*Es posible determinar, gestionar y predecir el comportamiento de los usuarios en diferentes plataformas web mediante el empleo*

*de minería de opiniones, técnicas de aprendizaje automático, métodos de recuperación de información y series temporales.*»

Una vez recordada nuestra hipótesis fundamental, exponemos las principales contribuciones que hemos logrado realizar gracias a esta investigación:

1. **Un modelo capaz de categorizar los comentarios publicados por los usuarios en el sitio web social *Menéame***. Este método de categorización de comentarios es capaz de clasificar un comentario, empleando técnicas de aprendizaje automático supervisado y de recuperación de información, en las distintas clases definidas con anterioridad: (i) tipo de comentario, (ii) enfoque del comentario y (iii) grado del comentario. De esta manera, hemos conseguido moderar un conjunto de comentarios descargado del sitio web social *Menéame*, con el fin de emplearlo como información complementaria en las siguientes etapas de la investigación. Este método puede ser empleado por administradores de páginas web con el fin de moderar sus sitios web sociales. Por ejemplo, puede ser usado para adecuar los comentarios y la visualización de la página acorde al usuario, para filtrar contenido que pueda dañar la imagen de marca de la página y también, para categorizar los usuarios vía sus comentarios. Además, a pesar de que nos hemos centrado en un sitio web social de noticias, dicho enfoque puede ser adaptado a cualquier página web que permita a sus usuarios el generar contenidos propios.

2. **Una representación de las características seleccionadas de los comentarios extraídos de *Menéame*.** Para poder moderar el conjunto de datos obtenido de dicho sitio web social es necesario la extracción de ciertas características, las cuales permitan posteriormente una categorización correcta de estos comentarios en cuestión. Para ello, las hemos dividido en 3 categorías diferentes: (i) características con propiedades estadísticas, (ii) características con propiedades sintácticas y (ii) características con propiedades de opinión. En el proceso, hemos utilizado técnicas de procesamiento del lenguaje natural y recuperación de la información.

3. **Un método basado en aprendizaje automático para clasificar usuarios de *Menéame* mediante su reputación.** Hemos desarrollado un procedimiento apto para la clasificación de usuarios del sitio web social *Menéame* en base a su reputación. Para ello, hemos empleado técnicas de aprendizaje automático, así como de recuperación y extracción de la información, con el fin de categorizar nuestro conjunto de usuarios y sus comentarios

en las siguientes clases: (i) tipo de usuario y (ii) grado del usuario. Únicamente con el conjunto de comentarios ya clasificado sería posible esta categorización, ya que para llevarla a cabo hemos adjuntado los comentarios a los usuarios correspondientes con el fin de obtener mayor información de éstos. Este método puede ser empleado por administradores de páginas web con el fin de moderar sus sitios web. Por ejemplo, puede ser usado para eliminar usuarios conflictivos, clasificar los usuarios según la publicidad en línea, etc.

4. **Un enfoque para la representación de las características extraídas de los usuarios seleccionados del sitio web social en cuestión.** Hemos extraído una serie de características del conjunto de usuarios, que unidas a las extraídas del conjunto de comentarios, permiten la correcta clasificación de los usuarios de *Menéame*. Estas características las hemos englobado en 2 categorías diferentes: (i) características extraídas del perfil y (ii) características extraídas de los comentarios. Para ello, hemos vuelto a emplear métodos de recuperación y extracción de información, así como el servicio *Google Imágenes*.

5. **Una mejora para el método de categorización de comentarios basado en técnicas de clasificación colectiva.** El problema con el aprendizaje automático supervisado es que se requiere un trabajo previo de etiquetado del conjunto de comentarios. Este proceso en el campo de la Web puede introducir una alta sobrecarga del rendimiento, debido al número de nuevos comentarios que se publican cada día por los usuarios del sitio web social. En esta tesis, hemos propuesto el primer método basado en aprendizaje colectivo para filtrar comentarios de usuarios trol del sistema, en base a características estadísticas, sintácticas y de opinión, que es capaz de determinar cuándo un comentario es polémico o no. Hemos validado empíricamente nuestro método empleando un conjunto de datos del sitio web social *Menéame*, mostrando como nuestra técnica, a pesar de necesitar un menor número de requisitos de etiquetado, obtiene la misma precisión que el aprendizaje supervisado.

6. **Una técnica basada en el aprendizaje colectivo capaz de mejorar el método de clasificación y reputación de usuarios pertenecientes a *Menéame*.** Uno de los problemas de los algoritmos de aprendizaje automático supervisado es que necesitan un trabajo previo para etiquetar el conjunto de datos de entrenamiento. Cuando se trata de la minería web, este proceso puede originar una alta sobrecarga en el rendimiento debido al número

de usuarios que publican comentarios en cualquier instante del día y a las personas que continuamente se crean nuevas cuentas en el sitio web social. En esta investigación, hemos propuesto el primer sistema basado en un método de filtrado de usuarios trol, en base a aprendizaje colectivo. Este enfoque es capaz de determinar cuándo un usuario es polémico o no, empleando las características del perfil público de los usuarios y las características estadísticas, sintácticas y de opinión de sus comentarios. Hemos validado empíricamente nuestro método utilizando un conjunto de datos de *Menéame*, mostrando que nuestra técnica obtiene la misma precisión que el aprendizaje supervisado, aún disponiendo de menor requerimientos de etiquetado.

7. **Un método basado en el análisis y predicción de series temporales para modelar el comportamiento de usuarios y sus comentarios en el sitio web social *Menéame*.** Gracias a las categorizaciones anteriores, tanto de comentarios como de usuarios, y al uso de las técnicas de análisis y predicción de series temporales, hemos podido modelar el comportamiento de los usuarios de *Menéame*. Con estos resultados logrados es posible la deducción del próximo comentario de un usuario y de esta manera poder tomar decisiones al respecto en caso de ser un comentario indebido.

8. **Una representación de las características necesarias para el empleo de series temporales a la hora de categorizar usuarios y sus comentarios.** Empleando conjuntamente las características extraídas de los comentarios y de los usuarios, hemos dividido éstas en 5 categorías: (i) características estadísticas, (ii) características sintácticas, (iii) características de opinión, (iv) características referentes al comentario y (v) características referentes al usuario. Con este grupo de características, hemos sido capaces de crear, y así poder analizar y predecir, las diferentes series temporales de los usuarios seleccionados.

Mediante el desarrollo de estas contribuciones, hemos estimado que los objetivos específicos descritos en la sección 1.4, y presentados a continuación, están cumplidos totalmente:

**Objetivo específico 1.** *Desarrollar un método para categorizar comentarios publicados por usuarios en sitios web sociales.*

**Objetivo específico 2.** *Diseñar y desarrollar nuevas técnicas para la clasificación y seguimiento de usuarios.*

**Objetivo específico 3.** *Implementar métodos eficaces para la gestión de la reputación de los usuarios y predicción de su comportamiento.*

Con lo cual, también hemos sido capaces de completar los objetivos operacionales presentados en la sección 1.4 y que hemos enumerado a continuación:

**Objetivo Operacional 1.** *Desarrollar un sistema para la categorización automática de comentarios en sitios web sociales.*

**Objetivo Operacional 2.** *Establecer las diferentes categorías para este sistema.*

**Objetivo Operacional 3.** *Implementar un nuevo método de categorización de usuarios.*

**Objetivo Operacional 4.** *Desarrollar una nueva técnica para el seguimiento de los usuarios.*

**Objetivo Operacional 5.** *Adecuación de los métodos de series temporales para la predicción temporal.*

**Objetivo Operacional 6.** *Realización de una prueba de concepto sobre predicción de acciones futuras de usuarios.*

**Objetivo Operacional 7.** *Generación de un modelo de reputación para los usuarios.*

Gracias al cumplimiento tanto de los objetivos específicos como de los objetivos operacionales, hemos podido ser capaces de satisfacer el objetivo principal de esta tesis doctoral:

**Objetivo principal 1.** *Desarrollar un sistema gestor y predictor de reputaciones de usuarios en sitios web sociales mediante la categorización automática de usuarios y sus comentarios.*

Por lo tanto, hemos considerado que esta investigación valida tanto la hipótesis fundamental como los correspondientes objetivos de esta tesis doctoral.

## 7.2 Limitaciones

Existen distintos temas a discutir en la presente investigación en referencia a los métodos propuestos. A continuación, hemos enumerado algunos de ellos.

El uso de algoritmos de aprendizaje automático supervisado para el entrenamiento del modelo, puede ser un problema en sí mismo. En nuestros experimentos, hemos empleado un conjunto de datos de entrenamiento de una semana. A medida que el tamaño conjunto de datos aumenta, también lo hace la cuestión de la escalabilidad. Este problema produce excesivos requerimientos de almacenamiento, incrementando la complejidad temporal perjudicando la precisión general de los modelos (Cano *et al*., 2006).

Nuestra técnica tiene varias limitaciones debido a la representación del texto. Por ejemplo, en el contexto del filtrado de *spam*, la mayoría de los métodos de filtrado están basados en las frecuencias con las que aparecen los términos en los mensajes y los usuarios que envían el *spam* han comenzado a modificar sus técnicas para evadir estos filtros. Estas técnicas pueden ser aplicadas por un usuario trol de un sitio web social de noticias.

Otro problema, es que solo generamos una visión estática del comportamiento de los usuarios. Se podría argumentar que los usuarios evolucionan continuamente y cambian la naturaleza de su comportamiento (por ejemplo, el envejecimiento, el matrimonio, etc.) Estas limitaciones de nuestra técnica implican la necesidad de encontrar métodos alternativos para la categorización dinámica de usuarios.

## 7.3 Trabajo futuro

### 7.3.1 Mejora en la reducción del conjunto de datos

Para reducir el almacenamiento desproporcionado y los costes de tiempo, es necesario reducir el conjunto de datos original (Czarnowski y Jedrzejowicz, 2006). Con el fin de resolver este problema, la reducción de datos es normalmente considerada una técnica apropiada de optimización del pre-procesado (Pyle, 1999; Tsang *et al*., 2003). Tales técnicas tienen muchas ventajas potenciales como la reducción de medición, almacenamiento y transmisión; la disminución de los tiempos de entrenamiento y testeo; el hacer frente a los *problemática asociada a la dimensionalidad* para mejorar el rendimiento de la predicción en términos de velocidad, precisión y sencillez y el facilitar la visualización y comprensión de los datos (Torkkola, 2003; Dash y Liu, 2003).

La reducción de datos puede ser implementada de 2 maneras. Por una parte, la selección de instancias (IS, de las voces inglesas *Instance Selection*) busca reducir el número de evidencias (por ejemplo, el número de filas) en el conjunto de datos de entrenamiento seleccionando las instancias más relevantes o remuestreando otras nuevas (Liu, 2010). Por otra parte, la selección de características (FS, del inglés *Feature Selection*) disminuye el número de atributos o características (por ejemplo, columnas) en el conjunto de datos de entrenamiento (Liu y Motoda, 2007).

Hemos aplicado FS en nuestros experimentos al seleccionar los atributos con una ganancia de información mayor que cero. Debido a que tanto el uso de IS como FS son muy efectivos reduciendo el tamaño del conjunto de datos de entrenamiento y ayudando a filtrar y limpiar ruido en los datos, mejorando así la precisión de los clasificadores de aprendizaje automático (Blum y Langley, 1997; Derrac *et al.*, 2009), recomendamos el empleo de estos métodos.

### 7.3.2 **La desambiguación del sentido de la palabra**

Hay un problema derivado del procesamiento del lenguaje natural (NLP, del inglés *Natural Language Processing*) cuando se trata de semántica: la desambiguación del sentido de la palabra (WSD, del inglés *Word Sense Disambiguation*). Un usuario trol puede evadir nuestro método intercambiando las palabras clave del comentario con otros términos polisémicos y así evitar la detección. De este modo, WSD se considera necesario para llevar a cabo la mayoría de las tareas de procesado de lenguaje natural (Ide y Véronis, 1998).

Por consiguiente, proponemos el estudio de diferentes técnicas de WSD (se puede encontrar un estudio sobre estas técnicas en Navigli (2009)) capaz de proporcionar una herramienta con reconocimiento semántico para la moderación. Sin embargo, tal enfoque semántico para la moderación debería tener que tratar con la semántica de los diversos lenguajes (Bates y Weischedel, 2006), y por lo tanto, ser dependiente del lenguaje.

### 7.3.3 **Mejora en la representación del texto**

Existen diferentes técnicas que pueden ser aplicadas por un usuario trol de un sitio web social de noticias. Por ejemplo, el método *Good Word Attack* modifica las estadísticas del término añadiendo un conjunto de palabras que son características legítimas, evitando así los filtros. No obstante, podemos adoptar algunos de los métodos que se han propuesto para mejorar el filtrado de *spam*, como

el aprendizaje multi-instancia (MIL, de las voces inglesas *Multiple Instance Learning*) (Dietterich *et al*., 1997). MIL divide una instancia o vector, en los métodos tradicionales de aprendizaje automático, en varias sub-instancias y categoriza el vector original basado en las sub-instancias (Maron y Lozano-Pérez, 1998).

Zhou *et al*. (2007) proponen la adopción del aprendizaje multi-instancia para el filtrado de *spam* dividiendo un email dentro de una *bolsa* de múltiples segmentos y clasificándolo como *spam*, si por lo menos una instancia en la correspondiente *bolsa* es *spam*. Por ello, podríamos adoptar este enfoque a nuestra técnica. Otro ataque, conocido como *tokenización*, trabaja en contra de la selección de características del comentario dividiendo o modificando las características clave del mensaje, que traduce la representación del término como no posible (Wittel y Wu, 2004). Todos estos ataques, los cuales los usuarios que envían el *spam* han venido adoptando, deberían tenerse en cuenta en la construcción de futuros sistemas de filtrado o moderación.

### 7.3.4 Evolución en el comportamiento del usuario

El método detección del tema y seguimiento (TDT, del inglés *Topic Detection and Tracking*) es una técnica que debe ser considerada. El método TDT asume múltiples fuentes de información y que la información proveniente de cada fuente es dividida en una secuencia de historias, que pueden proporcionar información sobre uno o más temas (o eventos) (Allan, 2002).

La tarea principal es identificar los eventos que se tratan en estas historias, en términos de las historias que los describen. Las historias que describen eventos inesperados seguirán el acontecimiento, mientras que las historias sobre eventos esperados pueden preceder y seguir al evento. Por estas razones, creemos que sería interesante el estudiar el método TDT en detalle para examinar su aplicabilidad a la categorización de comentarios y usuarios.

## 7.4 Consideraciones finales

La Web, tal y como la conocíamos, ha evolucionado a lo largo de los años. En estos momentos, no solo los administradores de los sitios web generan contenido. Los usuarios de los sitios web pueden generar sus propios contenidos, expresando sus opiniones o sentimientos respecto a ciertas noticias o eventos. Este hecho ha traído consigo consecuencias negativas como la aparición de contenido inapropiado u ofensivo, así como de los denominados usuarios trol. Normal-

mente, este tipo de usuarios pueden publicar también sus propios contenidos: *spam* o contenidos perjudiciales, dañando la integridad de otros usuarios.

Como muestra la Figura 7.1, el poder de penetración de Internet en todos nosotros es enorme[1]. En un país en el que el 41 % de la población dispone de una cuenta en un sitio web social, como es *Facebook*, se antoja primordial el poder controlar las acciones tomadas por ciertos usuarios que únicamente buscan crear polémica de manera deliberada.


**Figura 7.1:** Adaptación de la gráfica sobre los datos globales de la población española en Internet.

Por ello, esta investigación busca paliar este problema entre los distintos usuarios y las plataformas web, proponiendo un nuevo sistema de moderación automática y escalable que permita a las plataformas y a sus gestores disponer de una herramienta que aporte mayor información sobre dichos usuarios y su comportamiento. Por otra parte, los propios usuarios se verán más protegidos de contemplar contenido ofensivo, lo que inspirará mayor confianza y confortabilidad a la hora de navegar por el sitio web. Únicamente es una pequeña aportación para que este mundo funcione mejor entre todos nosotros, ya que...

*«Solo una vida vivida para los demás merece la pena ser vivida.»*

*Albert Einstein (1879–1955).*

[1]Fuente: We Are Social (`http://wearesocial.sg`)

*«Success consists of going from failure to failure without loss of enthusiasm.»*

Winston Leonard Spencer Churchill
(1874–1965)

CAPÍTULO

# 8

# Conclusions

THIS chapter concludes the present dissertation through the description of the established fundamental hypothesis and all the objectives, in order to evaluate their accomplishment thanks to the contributions described and performed throughout this doctoral investigation.

The remainder of this chapter is organised as follows. Section 8.1 describes the main contributions, whilst corroborates the fundamental hypothesis. Section 8.2 refers to the different limitations founded along the dissertation. Section 8.3 develops the future work and lines of research of the present work. Finally, section 8.4 exposes the final considerations of this thesis.

## 8.1 Main contributions

First, we have exposed the fundamental hypothesis described above, in order to obtain its corroboration through the accomplishment of all proposed objectives:

> «*It is possible to determine, manage and forecast the users' behaviour on diverse web platforms through opinion mining, machine learning techniques, information retrieval methods and time series.*»

Once remembered our fundamental hypothesis, we expose the main contributions of this dissertation:

1. **A model able to categorise published comments by users in the social news website *Menéame*.** This comment categorisation method is able to classify a comment in the different classes previously defined, using supervised machine-learning and information retrieval techniques: (i) type of information, (ii) focus of the comment and (iii) controversy level. In this way, we have moderated the gathered comment dataset from the social news website *Menéame*, in order to employ it as attached information in the following phases of the work. This method may be employed by administrators of webpages in order to moderate their website. For instance, it can be used to adequate the comments and visualisation of the page regarding the viewer, to filter content that may damage the brand image of the page and also to categorise the users via their comments. Besides, despite having focused on a social news site, such an approach can be adaptable to any webpage that allows its users to generate content for it.

2. **A selected features representation of the extracted comments from *Menéame*.** To moderate the dataset obtained from such social news website is necessary to extract certain features, which later allow a correct comment categorisation. For this, we have divided these features into 3 different categories: (i) statistical, (ii) syntactic and (ii) opinion features. In the process, we have utilised natural language processing and information retrieval techniques.

3. **A machine-learning-based method to classify *Menéame* users through their reputation.** We have developed a suitable procedure to classify users based on their reputation from the social news website *Menéame*. For this, we have employed machine-learning, information retrieval and information extraction techniques in order to categorise our user and their comments dataset in the following classes: (i) type of the user and (ii) controversy level of the user. Only with the comment dataset classified it would be possible to make this categorisation. We have attached the comments to the corresponding users in order to obtain more information about them. This method may be employed by administrators of webpages in order to moderate their website. For instance, it can be used to ban users, categorise the users for online advertising and so on.

4. **An approach for the representation of the extracted features from the selected users of the social new website *Menéame*.** We have extracted certain features of the user dataset, which together with the extracted features of the comment dataset, allow the *Menéame* users correct classification. We have divided these features into 2 different categories: (i) profile and (ii) comment related. To do this, we have used information extraction and information retrieval methods, as well as the *Google Images* service.

5. **An enhanced method for the comment categorisation based on collective classification techniques.** The problem with supervised learning is that a previous work of comments labelling is required. This process in the field of web can introduce a high performance overhead due to the number of new comments that appear everyday. In this dissertation, we have proposed the first collective-learning-based trolling comment filtering method system that based upon statistical, syntactic and opinion features, is able to determine when a comment is controversial or not. We have empirically validated our method using a dataset from *Menéame*, showing that our technique, despite having less labelling requirements, obtains the same accuracy than supervised learning.

6. **A semi-supervised-based technique able to improve the method to *Menéame* users classification and reputation.** One problem about supervised learning algorithms is that a preceding work is required to label the training dataset. When it comes to web mining, this process may originate a high performance overhead because of the number of users that post comments any time and the persons who create new accounts everyday. In this thesis, we have proposed the first trolling users filtering method system based on collective learning. This approach is able to determine when a user is controversial or not, employing users' profile features and statistical, syntactic and opinion features from their comments. We have empirically validated our method using a dataset from *Menéame*, showing that our technique obtains the same accuracy than supervised learning, despite having less labelling requirements.

7. **A method based on the time series analysis and forecasting to model the users' behaviour and their comments in the social news website *Menéame*.** Thanks to the previous comment and user categorisations and the utilisation of analysis and forecasting time series techniques, we have modelled the *Menéame* user behaviour. With the achieved results, it is pos-

sible to forecast the next user comment, and therefore analyse whether it will be inappropriate or not.

8. **A representation of the necessary features to employ time series in order to categorise users and their comments.** Using together the extracted features from the comments and users, we have divided these features into 5 categories: (i) statistical features, (ii) syntactic features, (iii) opinion features, (iv) comment related features and (v) user related features. With this set of features, we have been able to create, analyse and forecast the diverse user time series selected.

Through the development of these contributions, we have estimated that the specific goals described in section 1.4 are totally accomplished. Next, we present them:

**Specific goal 1.** *Develop a method for categorising comments published by users on social news websites.*

**Specific goal 2.** *Design and develop new techniques for classifying and tracking users.*

**Specific goal 3.** *Implement effective methods to manage the users' reputation and predict their behaviour.*

Also, we were able to fulfil the operational goals presented in the previous section 1.4:

**Operational goal 1.** *Develop a system in order to categorise automatically comments in social news websites.*

**Operational goal 2.** *Establish the diverse categories for this system.*

**Operational goal 3.** *Implement a new user categorisation method.*

**Operational goal 4.** *Develop a new technique for tracking users.*

**Operational goal 5.** *Adequate time series methods to temporal prediction.*

**Operational goal 6.** *Realise a concept test about the future user actions prediction.*

**Operational goal 7.** *Generate a reputation user model.*

Thanks to the achievement of all the goals (operational and specific), we were able to accomplish the main goal of this dissertation:

**Main goal 1.** *Develop a system able to manage and forecast users' behaviours in social news websites by automatically categorisation of users and their comments.*

Therefore, we consider that this investigation validates the fundamental hypothesis and the corresponding goals of this dissertation.

## 8.2 Limitations

There are several important points to be discussed referring to the appropriateness of our proposed methods. Next we introduce some of them.

The use of supervised machine-learning algorithms for the model training, can be a problem itself. In our experiments, we used a training dataset of one week. As the dataset size grows, so does the issue of scalability. This problem produces excessive storage requirements, increases time complexity and impairs the general accuracy of the models (Cano *et al.*, 2006).

Our technique has several limitations due to the representation of text. For instance, in the context of spam filtering, most of the filtering techniques are based on the frequency of occurrence of terms within messages. Therefore, spammers have started modifying their techniques to evade such filters. These techniques can be applied by a troll user of a social news website.

Another problem, is that we only generate a static view of the users' behaviour. One may argue that users continuously evolve and change the nature of their behaviour (e.g., ageing, getting married, etc.). These limitations of our technique imply the need to find alternative methods for a dynamic user categorisation.

## 8.3 Future work

### 8.3.1 Enhancing the dataset reduction

To reduce disproportionate storage and time costs, it is necessary to reduce the original training set (Czarnowski y Jedrzejowicz, 2006). In order to solve this

issue, *data reduction* is normally considered an appropriate preprocessing optimisation technique (Pyle, 1999; Tsang *et al.*, 2003). Such techniques have many potential advantages such as reducing measurement, storage and transmission; decreasing training and testing times; confronting the *curse of dimensionality* to improve prediction performance in terms of speed, accuracy and simplicity and facilitating data visualization and understanding (Torkkola, 2003; Dash y Liu, 2003).

Data reduction can be implemented in two ways. On the one hand, *Instance Selection* (IS) seeks to reduce the evidences (i.e., number of rows) in the training set by selecting the most relevant instances or re-sampling new ones (Liu, 2010). On the other hand, *Feature Selection* (FS) decreases the number of attributes or features (i.e., columns) in the training set (Liu y Motoda, 2007).

We applied FS in our experiments when selecting the attributes with more than zero of Information Gain. Because both IS and FS are very effective at reducing the size of the training set and helping to filtrate and clean noisy data, thereby improving the accuracy of machine-learning classifiers (Blum y Langley, 1997; Derrac *et al.*, 2009), we strongly encourage the use of these methods.

### 8.3.2 Word sense disambiguation

There is an issue derived from Natural Language Processing (NLP) when dealing with semantics: *Word Sense Disambiguation* (WSD). A troll user may evade our method by explicitly exchanging the key words of the comment with other polyseme terms and thus avoid detection. Thereby, WSD is considered necessary to perform most natural language processing tasks (Ide y Véronis, 1998).

Hence, we propose the study of different WSD techniques (a survey of different WSD techniques can be found in Navigli (2009)) able to provide a semantics-aware moderation tool. However, such a semantic approach for moderation would have to deal with the semantics of different languages (Bates y Weischedel, 2006) and, therefore, it should be language dependant.

### 8.3.3 Improving the text representation

There are some techniques that can be applied by a troll user of a social news website. For example, *Good Word Attack* is a method that modifies the term statistics by appending a set of words that are characteristic of legitimate, thereby bypassing filters. Nevertheless, we can adopt some of the methods that

have been proposed in order to improve spam filtering, such as *Multiple Instance Learning* (MIL) (Dietterich *et al*., 1997). MIL divides an instance or a vector in the traditional supervised learning methods into several sub-instances and classifies the original vector based on the sub-instances (Maron y Lozano-Pérez, 1998).

Zhou *et al*. (2007) proposed the adoption of multiple instance learning for spam filtering by dividing an email into a bag of multiple segments and classifying it as spam if at least one instance in the corresponding bag was spam. We can adapt this approach to our comment moderation tool. Another attack, known as *tokenisation*, works against the feature selection of the comment by splitting or modifying key message features, which renders the term representation as no longer feasible (Wittel y Wu, 2004). All of these attacks, which spammers have been adopting, should be taken into account in the construction of future filtering or moderation systems.

### 8.3.4 **Evolving user's behaviour**

The Topic Detection and Tracking (TDT) method is a technique that may be considered. The TDT method assumes multiple sources of information and that the information owing from each source is divided into a sequence of stories, which may provide information on one or more topics (or events) (Allan, 2002).

The general task is to identify the events being discussed in these stories, in terms of the stories that describe them. Stories that describe unexpected events will obviously follow the event, whereas stories on expected events can both precede and follow the event. For these reasons, we believe it would be interesting to study the TDT method in detail to examine its applicability to user and comment categorisation.

## 8.4 **Final remarks**

The Web has evolved over the years. Now, not only the administrators of a site generate content. Users of a website can make content available expressing their opinions or sentiments regarding to certain news or events. This fact has led to some negative side effects: the content generated is inappropriate and the emergence of troll users. Frequently, this type of users can also publish: spam or harmful content, hurting other web users.

The Figure 8.1 shows the huge Internet power penetration in all of us[1]. In a country where 41 % of the population has an account on a social website, as Facebook, it seems essential to control the actions taken by certain users who deliberately seek controversy.

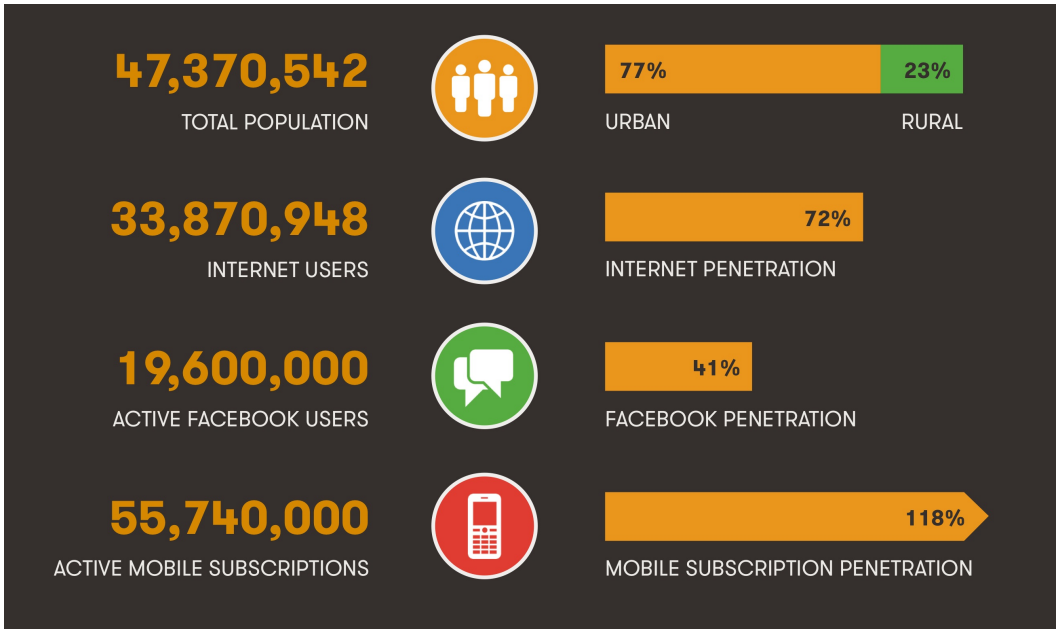



**Figura 8.1:** Adaptation graph about the global data of Spanish population in Internet.

Therefore, this dissertation seeks to mitigate this problem between users and web platforms. We propose a new automatic and scalable moderation system allowing the platforms and their administrators to dispose a system that provides more information about the users and their behaviour. On the other hand, such users will also be better protected against being exposed to offensive content. This fact inspire greater confidence and comfort when navigating the social website. It is only a small contribution to make a better world all together, because...

*«Only a life lived for others is a life worth living.»*

*Albert Einstein (1879–1955).*

[1]Fuente: We Are Social (`http://wearesocial.sg`)

# Publicaciones

Algunas de las ideas o de las figuras que aparecen en esta tesis han aparecido previamente en las siguientes publicaciones:

## Artículos en revistas internacionales

1. I. Santos, J. de-la-Peña-Sordo, I. Pastor-López, P. Galán-García y P. G. Bringas. (2012). *Automatic categorisation of comments in social news websites*. Expert Systems with Applications, 39(18), pp. 13417-13425. ISBN: 0957-4174. Factor de impacto: 1,854 (JCR 2012) - 1º cuartil. DOI: `http://dx.doi.org/10.1016/j.eswa.2012.05.061`.

## Artículos en conferencias internacionales

2. J. de-la-Peña-Sordo, I. Santos y P. G. Bringas. (2013). *Moderación Automática de Comentarios y Reputación de Usuarios en Sitios Web Sociales*. En las actas de la 15ª Conferencia de la Asociación Española para la Inteligencia Artificial (CAEPIA'13), Madrid (España), 17-20 de Septiembre de 2013, ISBN: 978-84-695-8348-7, pp. 1618-1623.

3. J. de-la-Peña-Sordo, I. Santos, I. Pastor-López y P. G. Bringas. (2013). *Social News Website Moderation through Semi-supervised Troll User Filtering*. En las actas de la 6ª Conferencia Internacional en Computational Intelligence in Security for Information Systems (CISIS 2013) Joint Conference SOCO'13-CISIS'13-ICEUTE'13, Salamanca (España), 11-13 de Septiembre de 2013, ISBN en línea: 978-3-319-01854-6, ISBN impresión: 978-3-319-01853-9, Volumen series: 239, ISSN series: 2194-5357, pp. 577-587, Springer International Publishing. DOI: `http://dx.doi.org/10.1007/978-3-319-01854-6_59`.

4. J. de-la-Peña-Sordo, I. Santos, I. Pastor-López y P. G. Bringas. (2013). *Filtering Trolling Comments through Collective Classification*. En las actas de la 7ª Conferencia Internacional en Network and System Security (NSS 2013), Madrid (España), 3-4 de Junio de 2013, ISBN en línea: 978-3-642-38631-2, ISBN impresión: 978-3-642-38630-5, Volumen series: 7873, ISSN series: 0302-9743, pp. 707-713, Springer Berlin Heidelberg. DOI: http://dx.doi.org/10.1007/978-3-642-38631-2_60.

Algunas de las ideas o de las figuras relacionadas directamente con esta investigación han aparecido previamente en las siguientes publicaciones:

## Artículos en conferencias internacionales

1. J. de-la-Peña-Sordo, I. Pastor-López, X. Ugarte-Pedrero, I. Santos y P. G. Bringas. (2014). *Anomalous User Comment Detection in Social News Websites*. En las actas de la 7ª Conferencia Internacional en Computational Intelligence in Security for Information Systems (CISIS 2014) Joint Conference SOCO'14-CISIS'14-ICEUTE'14, Bilbao (España), 25-27 de Junio de 2014, ISBN en línea: 978-3-319-07995-0, ISBN impresión: 978-3-319-07994-3, Volumen series: 299, ISSN series: 2194-5357, pp. 517-526, Springer International Publishing. DOI: http://dx.doi.org/10.1007/978-3-319-07995-0_51 .

Otras publicaciones en las que participa el autor de esta tesis doctoral son enumeradas a continuación:

## Artículos en revistas internacionales

1. I. Pastor-López, J. de-la-Peña-Sordo, S. Rojas, I. Santos y P. G. Bringas. (2014). *Visión artificial basada en aprendizaje automático para la categorización de defectos superficiales en fundición*. Dyna, 89(3), pp. 325-332. ISSN: 0012-7361. Factor de impacto: 0,237 (JCR 2012) - 4º cuartil. DOI: http://dx.doi.org/10.6036/6940.

2. J. de-la-Peña-Sordo, I. Ruiz-Agundez, J. López-de-Uralde y P. G. Bringas. (2011). *El proyecto ZIURFONE hace más seguras nuestras comunicaciones de VoIP.*. Deusto Ingeniería, 12, pp. 56-58.

## Artículos en conferencias internacionales

3. I. Pastor-López, I. Santos, J. de-la-Peña-Sordo, I. García-Ferreira, A. G. Zabala y P. G. Bringas. (2014). *Enhanced Image Segmentation Using Quality Threshold Clustering for Surface Defect Categorisation in High Precision Automotive Castings*. En las actas de la 9ª Conferencia Internacional en Soft Computing Models in Industrial and Environmental Applications (SOCO 2013) Joint Conference SOCO'13-CISIS'13-ICEUTE'13, Salamanca (España), 11-13 de Septiembre de 2013, ISBN en línea: 978-3-319-01854-6, ISBN impresión: 978-3-319-01853-9, Volumen series: 239, ISSN series: 2194-5357, pp. 191-200, Springer International Publishing. DOI: `http://dx.doi.org/10.1007/978-3-319-01854-6_20`.

4. I. Pastor-López, I. Santos, J. de-la-Peña-Sordo, M. Salazar, A. Santamaria-Ibirika y P. G. Bringas. (2013). *Collective classification for the detection of surface defects in automotive castings*. En las actas de la 8ª Conferencia IEEE en Industrial Electronics and Applications (ICIEA 2013), Melbourne (Australia), 19-21 de Junio de 2013, ISBN impresión: 978-1-4673-6320-4, pp. 941-946, IEEE. DOI: `http://dx.doi.org/10.1109/ICIEA.2013.6566502`.

5. I. Pastor-López, I. Santos, A. Santamaria-Ibirika, M. Salazar, J. de-la-Peña-Sordo y P. G. Bringas. (2012). *Machine-learning-based surface defect detection and categorisation in high-precision foundry.* En las actas de la 7ª Conferencia IEEE en Industrial Electronics and Applications (ICIEA 2012), Singapur (Singapur), 18-20 de Julio de 2012, ISBN impresión: 978-1-4577-2118-2, pp. 1359-1364, IEEE. DOI: `http://dx.doi.org/10.1109/ICIEA.2012.6360934`.

# Bibliografía


Abdul-Rahman, A. y Hailes, S. (2000). Supporting trust in virtual communities. En *System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on*, páginas 9–pp. IEEE.

Allan, J. (2002). Introduction to topic detection and tracking. En *Topic detection and tracking*, páginas 1–16. Springer.

Amari, S. y Wu, S. (1999). Improving support vector machine classifiers by modifying kernel functions. *Neural Networks*, 12(6):783–789.

Anders, U. y Korn, O. (1999). Model selection in neural networks. *Neural Networks*, 12(2):309–323.

Baeza-Yates, R., Ribeiro-Neto, B., *et al*. (1999). *Modern information retrieval*, volumen 463. ACM press New York.

Baeza-Yates, R. A. y Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Baker, P. (2001). Moral panic and alternative identity construction in usenet. *Journal of Computer-Mediated Communication*, 7(1):0–0.

Bates, M. y Weischedel, R. M. (2006). *Challenges in natural language processing*. Cambridge University Press.

Belousov, A., Verzakov, S., y Von Frese, J. (2002). A flexible classification approach with optimal generalisation performance: support vector machines. *Chemometrics and Intelligent Laboratory Systems*, 64(1):15–25.

Bergstrom, K. (2011). "don't feed the troll": Shutting down debate about community expectations on reddit.com. *First Monday*, 16(8).

Berners-Lee, T., Hendler, J., Lassila, O., *et al*. (2001). The semantic web. *Scientific american*, 284(5):28–37.

Berthold, M. y Hand, D. I. (2000). Intelligent data analysis. *Technometrics*, 42(4):442–442.

Bertino, E., Ferrari, E., y Perego, A. (2006). Web content filtering. *Web and Information Security*, páginas 112–132.

Berwick, R. C., Abney, S. P., y Tenny, C. (1991). *Principle-Based Parsing: Computation and Psycholinguistics*, volumen 44. Springer.

Beveridge, W. H. (1922). Wheat prices and rainfall in western europe. *Journal of the Royal Statistical Society*, 85(3):412–475.

Bishop, C. M. (1995). *Neural Networks for Pattern Recognition*. Oxford University Press.

Blum, A. L. y Langley, P. (1997). Selection of relevant features and examples in machine learning. *Artificial intelligence*, 97(1):245–271.

Bose, I. y Raktim, P. (2005). Using support vector machines to evaluate financial fate of dotcoms. *PACIS 2005 Proceedings. Paper*, 42.

Box, G. y Jenkins, G. (1970). Time series analysis: Forecasting and control. *Holden-D. iv, San Francisco*.

Box, G. E., Jenkins, G. M., y Reinsel, G. C. (2011). *Time series analysis: forecasting and control*, volumen 734. Wiley.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Burges, C. J. (1998). A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, 2(2):121–167.

Burgess, J. y Green, J. (2009). *YouTube: Online video and participatory culture*. Polity press.

Burns, A. F. y Mitchell, W. C. (1946). Measuring business cycles. *NBER Books*.

Buys-Ballot, C. H. D. (1847). Les changements périodiques de température. *Kemink et Fils, Utrecht*.

Cahill, V., Gray, E., Seigneur, J.-M., Jensen, C. D., Chen, Y., Shand, B., Dimmock, N., Twigg, A., Bacon, J., English, C., *et al*. (2003). Using trust for secure collaboration in uncertain environments. *Pervasive Computing, IEEE*, 2(3):52–61.

Cambria, E., Chandra, P., Sharma, A., y Hussain, A. (2010). Do not feel the trolls. *the 9th ISWC*.

Cano, J. R., Herrera, F., y Lozano, M. (2006). On the combination of evolutionary algorithms and stratified strategies for training set selection in data mining. *Applied Soft Computing*, 6(3):323–332.

Cao, L. (2003). Support vector machines experts for time series forecasting. *Neurocomputing*, 51:321–339.

Carbone, M., Nielsen, M., y Sassone, V. (2003). A formal model for trust in dynamic networks. En *Software Engineering and Formal Methods, 2003. Proceedings. First International Conference on*, páginas 54–61. IEEE.

Castillo, E., Gutiérrez, J., y Hadi, A. (1997). *Expert systems and probabilistic network models*. Springer Verlag.

Chalimourda, A., Schölkopf, B., y Smola, A. J. (2004). Experimentally optimal $\nu$ in support vector regression for different noise models and parameter settings. *Neural Networks*, 17(1):127–141.

Chao, J. (2007). Student project collaboration using wikis. En *Software Engineering Education & Training, 2007. CSEET'07. 20th Conference on*, páginas 255–261. IEEE.

Chapelle, O., Schölkopf, B., Zien, A., *et al*. (2006). *Semi-supervised learning*, volumen 2. MIT press Cambridge, MA.

Chawla, N., Bowyer, K., Hall, L., y Kegelmeyer, W. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16(3):321–357.

Chen, M.-S., Han, J., y Yu, P. S. (1996). Data mining: an overview from a database perspective. *Knowledge and data Engineering, IEEE Transactions on*, 8(6):866–883.

Cherkassky, V. y Ma, Y. (2004). Practical selection of svm parameters and noise estimation for svm regression. *Neural networks*, 17(1):113–126.

Cho, B., Yu, H., Lee, J., Chee, Y., Kim, I., y Kim, S. (2008). Nonlinear support vector machine visualization for risk factor analysis using nomograms and localized radial basis function kernels. *IEEE Transactions on Information Technology in Biomedicine*, 12(2):247–256.

Clausen, A. (2004). The cost of attack of pagerank. En *Proceedings of The International Conference on Agents, Web Technologies and Internet Commerce (IAWTIC'2004)*.

Coleman, B. (2010). Hacker and troller as trickster. *online] tilgået d*, 6(5):2012.

Cooper, G. y Herskovits, E. (1991a). A bayesian method for constructing bayesian belief networks from databases. En *Proceedings of the Seventh Conference on Uncertainty in Artificial Intelligence*, páginas 86–94. Morgan Kaufmann Publishers Inc.

Cooper, G. F. y Herskovits, E. (1991b). A bayesian method for constructing bayesian belief networks from databases. En *Proceedings of the 1991 conference on Uncertainty in artificial intelligence*.

Corona, R., Sánchez, R., Schiavon, M., y Einstein, A. (2012). Metodología para la detección de cuentas trolls en twitter el caso de las elecciones presidenciales de 2012.

Cortes, C. y Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.

Cottrell, M., Girard, B., Girard, Y., Mangeas, M., y Muller, C. (1995). Neural modeling for time series: a statistical stepwise method for weight elimination. *Neural Networks, IEEE Transactions on*, 6(6):1355–1364.

Croft, W. B., Metzler, D., y Strohman, T. (2010). *Search engines: Information retrieval in practice*. Addison-Wesley.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems (MCSS)*, 2(4):303–314.

Czarnowski, I. y Jedrzejowicz, P. (2006). Instance reduction approach to machine learning and multi-database mining. En *Proceedings of the Scientific Session organized during XXI Fall Meeting of the Polish Information Processing Society, Informatica, ANNALES Universitatis Mariae Curie-Skłodowska, Lublin*, páginas 60–71. Citeseer.

Dalkey, N. y Helmer, O. (1963). An experimental application of the delphi method to the use of experts. *Management science*, 9(3):458–467.

Darbellay, G. A. y Slama, M. (2000). Forecasting the short-term demand for electricity: Do neural networks stand a better chance? *International Journal of Forecasting*, 16(1):71–83.

Dash, M. y Liu, H. (2003). Consistency-based search in feature selection. *Artificial intelligence*, 151(1):155–176.

De Groot, C. y Wuertz, D. (1991). Analysis of univariate time series with connectionist nets: A case study of two classical examples. *Neurocomputing*, 3(4):177–192.

deGroote, E. (2005). Machine learning in go: Supervised learning in move prediction.

Dempster, A. P., Laird, N. M., y Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, páginas 1–38.

Derrac, J., García, S., y Herrera, F. (2009). A first study on the use of coevolutionary algorithms for instance and feature selection. En *Hybrid Artificial Intelligence Systems*, páginas 557–564. Springer.

Dietterich, T. G., Lathrop, R. H., y Lozano-Pérez, T. (1997). Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1):31–71.

Donath, J. *et al*. (1999). Identity and deception in the virtual community. *Communities in cyberspace*, 1996:29–59.

Elahi, S., d'Aquin, M., y Motta, E. (2010). Who wants a piece of me? reconstructing a user profile from personal web activity logs. En *International ESWC Workshop on Linking of User Profiles and Applications in the Social Semantic Web*.

Ellison, N. *et al*. (2007). Social network sites: Definition, history, and scholarship. *Journal of Computer-Mediated Communication*, 13(1):210–230.

Faraway, J. y Chatfield, C. (1998). Time series forecasting with neural networks: a comparative study using the air line data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 47(2):231–250.

Fayyad, U., Piatetsky-Shapiro, G., Smyth, P., *et al*. (1996). Knowledge discovery and data mining: Towards a unifying framework. *Knowledge Discovery and Data Mining*, páginas 82–88.

Feldman, R. y Dagan, I. (1995). Knowledge discovery in textual databases (kdt). En *Proc. 1st Int. Conf. Knowledge Discovery and Data Mining*, páginas 112–117.

Fisher, D. H. (1987). Knowledge acquisition via incremental conceptual clustering. *Machine learning*, 2(2):139–172.

Fix, E. y Hodges Jr, J. L. (1952). Discriminatory analysis-nonparametric discrimination: Small sample performance. Technical report, DTIC Document.

Freeman, L. C. (1979). Centrality in social networks conceptual clarification. *Social networks*, 1(3):215–239.

Friedman, J., Hastie, T., y Tibshirani, R. (2001). *The elements of statistical learning*, volumen 1. Springer Series in Statistics.

Friedman, N., Geiger, D., y Goldszmidt, M. (1997). Bayesian network classifiers. *Machine learning*, 29(2):131–163.

Fumero, A. (2005). Un tutorial sobre blogs. el abecé del universo blog. *Telos: Cuadernos de comunicación e innovación*, (65):46–59.

Funahashi, K.-I. (1989). On the approximate realization of continuous mappings by neural networks. *Neural networks*, 2(3):183–192.

García-Jiménez, B., Ledezma, A., y Sanchis, A. (2011). Mmrf for proteome annotation applied to human protein disease prediction. En *Inductive Logic Programming*, páginas 67–75. Springer.

Garner, S. R. *et al*. (1995). Weka: The waikato environment for knowledge analysis. En *Proceedings of the New Zealand computer science research students conference*, páginas 57–64. Citeseer.

Geiger, D., Goldszmidt, M., Provan, G., Langley, P., y Smyth, P. (1997). Bayesian network classifiers. En *Machine Learning*, páginas 131–163.

Gómez Hidalgo, J. M., Sanz, E. P., García, F. C., y Rodríguez, M. D. B. (2009). Web content filtering. *Advances in Computers*, 76:257–306.

Górriz, J., Puntonet, C. G., y Salmerón, M. (2004a). A comparison between exogenous methods for times series prediction.

Górriz, J., Puntonet, C. G., Salmerón, M., Ortega, J., y Aldasht, M. (2004b). Time series forecasting based on parallel neural network.

Granger, C. W. y Terasvirta, T. (2011). Modelling non-linear economic relationships. *OUP Catalogue*.

Gruber, T. (2007). Ontology of folksonomy: A mash-up of apples and oranges. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 3(1):1–11.

Han, J. y Kamber, M. (2006). *Data mining: concepts and techniques*. Morgan Kaufmann.

Hardaker, C. (2010). Trolling in asynchronous computer-mediated communication: From user discussions to academic definitions. *Journal of Politeness Research. Language, Behaviour, Culture*, 6(2):215–242.

Hastie, T., Tibshirani, R., Friedman, J., y Franklin, J. (2005). The elements of statistical learning: data mining, inference and prediction. *The Mathematical Intelligencer*, 27(2):83–85.

Hearst, M. A. (1999). Untangling text data mining. En *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, páginas 3–10. Association for Computational Linguistics.

Heravi, S., Osborn, D. R., y Birchenhall, C. (2004). Linear versus neural network forecasts for european industrial production series. *International Journal of Forecasting*, 20(3):435–446.

Herring, S., Job-Sluder, K., Scheckler, R., y Barab, S. (2002). Searching for safety online: Managing "trolling" in a feminist forum. *The Information Society*, 18(5):371–384.

Hidalgo, J. G., Sanz, E. P., García, F. C., y deBuenaga Rodríguez, M. (2003). Categorización de texto sensible al coste para el filtrado de contenidos inapropiados en internet. *Procesamiento del lenguaje natural*, 31:13–20.

Holmes, G., Donkin, A., y Witten, I. H. (1994). Weka: A machine learning workbench. En *Intelligent Information Systems, 1994. Proceedings of the 1994 Second Australian and New Zealand Conference on*, páginas 357–361. IEEE.

Holt, S. (1957). A method for determining gear selectivity and its application. *Joint Sci. Meet. ICNAF, ICES, and FAO, Pap*, (515).

Hornik, K., Stinchcombe, M., y White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366.

Hsu, C.-W., Chang, C.-C., y Lin, C.-J. (2003). A practical guide to support vector classification.

Hung, K. K. y Xu, L. (1999). Further study of adaptive supervised learning decision (asld) network in stock market. En *IEEE-EURASIP Workshop Nonlinear Signal Image Processing*. Citeseer.

Ide, N. y Véronis, J. (1998). Introduction to the special issue on word sense disambiguation: the state of the art. *Computational linguistics*, 24(1):2–40.

Idika, N. y Mathur, A. P. (2007). A survey of malware detection techniques. *Purdue University*, página 48.

Jindal, N. y Liu, B. (2007). Review spam detection. En *Proceedings of the 16th international conference on World Wide Web*, páginas 1189–1190. ACM.

Jindal, N. y Liu, B. (2008). Opinion spam and analysis. En *Proceedings of the international conference on Web search and web data mining*, páginas 219–230.

Joachims, T. y Radlinski, F. (2007). Search engines that learn from implicit feedback. *Computer*, 40(8):34–40.

Jones, K. S. (1997). *Readings in information retrieval*. Morgan Kaufmann.

Jøsang, A. y Pope, S. (2005). Semantic constraints for trust transitivity. En *Proceedings of the 2nd Asia-Pacific conference on Conceptual modelling-Volume 43*, páginas 59–68. Australian Computer Society, Inc.

Jsang, A. y Ismail, R. (2002). The beta reputation system. En *Proceedings of the 15th bled electronic commerce conference*, páginas 41–55.

Kaastra, I. y Boyd, M. (1996). Designing a neural network for forecasting financial and economic time series. *Neurocomputing*, 10(3):215–236.

Kalman, R. E. *et al*. (1960). A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45.

Kamvar, S. D., Schlosser, M. T., y Garcia-Molina, H. (2003). The eigentrust algorithm for reputation management in p2p networks. En *Proceedings of the 12th international conference on World Wide Web*, páginas 640–651. ACM.

Kent, J. T. (1983). Information gain and a general measure of correlation. *Biometrika*, 70(1):163–173.

Kiesler, S., Siegel, J., y McGuire, T. (1984). Social psychological aspects of computer-mediated communication. *American psychologist*, 39(10):1123.

Kim, K.-j. (2003). Financial time series forecasting using support vector machines. *Neurocomputing*, 55(1):307–319.

Kim, S.-M. y Hovy, E. (2007). Crystal: Analyzing predictive opinions on the web. En *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, páginas 1056–1064.

Kleinberg, E. (1996). An overtraining-resistant stochastic modeling method for pattern recognition. *The annals of statistics*, 24(6):2319–2349.

Kodratoff, Y. (1999). Knowledge discovery in texts: a definition, and applications. *Foundations of Intelligent Systems*, páginas 16–29.

Kolmogorov, A. N. (1941). Dissipation of energy in locally isotropic turbulence. En *Dokl. Akad. Nauk SSSR*, volumen 32, páginas 16–18.

Kuan, C.-M. y Liu, T. (1995). Forecasting exchange rates using feedforward and recurrent neural networks. *Journal of Applied Econometrics*, 10(4):347–364.

Lagoze, C. y Hunter, J. (2001). The abc ontology and model. En *DC-2001: International Conference on Dublin Core and Metadata Applications 2001*, volumen 2, páginas 1–18. British Computer Society and Oxford University Press.

Laorden, C., Sanz, B., Santos, I., Galán-García, P., y Bringas, P. G. (2011). Collective classification for spam filtering. En *Computational Intelligence in Security for Information Systems*, páginas 1–8. Springer.

Lea, M., O'Shea, T., Fung, P., y Spears, R. (1992). *"Flaming" in computer-mediated communication: Observations, explanations, implications.* Harvester Wheatsheaf.

Lerman, K. (2007). User participation in social media: Digg study. En *Web Intelligence and Intelligent Agent Technology Workshops, 2007 IEEE/WIC/ACM International Conferences on*, páginas 255–258. IEEE.

Leshno, M., Lin, V. Y., Pinkus, A., y Schocken, S. (1993). Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural networks*, 6(6):861–867.

Levenberg, K. (1944). A method for the solution of certain problems in least squares. *Quarterly of applied mathematics*, 2:164–168.

Levien, R. L. (2002). *Attack resistant trust metrics*. PhD thesis, Citeseer.

Lewis, D. (1998). Naive (bayes) at forty: The independence assumption in information retrieval. *Machine Learning: ECML-98*, páginas 4–15.

Liu, H. (2010). *Instance selection and construction for data mining*. Springer-Verlag.

Liu, H. y Motoda, H. (2007). *Computational methods of feature selection*. Chapman and Hall/CRC.

Lloyd, S. (1982). Least squares quantization in pcm. *Information Theory, IEEE Transactions on*, 28(2):129–137.

Lu, C.-J., Lee, T.-S., y Chiu, C.-C. (2009). Financial time series forecasting using independent component analysis and support vector regression. *Decision Support Systems*, 47(2):115–125.

Maitra, R. (2002). A statistical perspective on data mining. *J. Ind. Soc. Prob. Statist*.

Maji, S., Berg, A., y Malik, J. (2008). Classification using intersection kernel support vector machines is efficient. En *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, páginas 1–8. IEEE.

Makridakis, S., Wheelwright, S. C., y Hyndman, R. J. (2008). *Forecasting methods and applications*. John Wiley & Sons.

Manchala, D. W. (1998). Trust metrics, models and protocols for electronic commerce transactions. En *Distributed Computing Systems, 1998. Proceedings. 18th International Conference on*, páginas 312–321. IEEE.

Maron, O. y Lozano-Pérez, T. (1998). A framework for multiple-instance learning. *Advances in neural information processing systems*, páginas 570–576.

Marsden, P. y Lin, N. (1982). *Social structure and networks analysis*. Número 57. Beverley Hills [etc.]: SAGE.

Masters, T. (1993). *Practical neural network recipes in C++*. Morgan Kaufmann.

Masters, T. (1995). *Neural, novel and hybrid algorithms for time series prediction*. John Wiley & Sons, Inc.

Mayrhofer, R., Radi, H., y Ferscha, A. (2003). Recognizing and predicting context by learning from user behavior. En *The International Conference On Advances in Mobile Multimedia (MoMM2003)*, volumen 171, páginas 25–35.

Mcnamee, P. y Mayfield, J. (2004). Character n-gram tokenization for european language text retrieval. *Information Retrieval*, 7(1):73–97.

McNamee, P. y Mayfield, J. (2004). Jhu/apl experiments in tokenization and non-word translation. *Comparative Evaluation of Multilingual Information Access Systems*, páginas 85–97.

Medeiros, M. C., Teräsvirta, T., y Rech, G. (2006). Building neural network models for time series: a statistical approach. *Journal of Forecasting*, 25(1):49–75.

Mitchell, T. M. (1997). Machine learning. 1997. *Burr Ridge, IL: McGraw Hill*.

Mohan, V. V. (2009). *Analysis and Prediction of Web User Interactions Using Time Series Analysis*. PhD thesis, The Pennsylvania State University.

Moody, J. y Darken, C. J. (1989). Fast learning in networks of locally-tuned processing units. *Neural computation*, 1(2):281–294.

Mui, L., Mohtashemi, M., y Ang, C. (2001a). A probabilistic rating framework for pervasive computing environments. En *Proceedings of the MIT Student Oxygen Workshop (SOW'2001)*.

Mui, L., Mohtashemi, M., Ang, C., Szolovits, P., y Halberstadt, A. (2001b). Ratings in distributed systems: A bayesian approach. En *Proceedings of the Workshop on Information Technologies and Systems (WITS)*, páginas 1–7.

Mui, L., Mohtashemi, M., y Halberstadt, A. (2002). A computational model of trust and reputation. En *System Sciences, 2002. HICSS. Proceedings of the 35th Annual Hawaii International Conference on*, páginas 2431–2439. IEEE.

Mukherjee, S., Osuna, E., y Girosi, F. (1997). Nonlinear prediction of chaotic time series using support vector machines. En *Neural Networks for Signal Processing [1997] VII. Proceedings of the 1997 IEEE Workshop*, páginas 511–520. IEEE.

Müller, K., Smola, A., Rätsch, G., Schölkopf, B., Kohlmorgen, J., y Vapnik, V. (1997). Predicting time series with support vector machines. *Artificial Neural Networks—ICANN'97*, páginas 999–1004.

Namata, G., Sen, P., Bilgic, M., y Getoor, L. (2009). Collective classification for text classification. *Text Mining*, páginas 51–69.

Nasraoui, O., Soliman, M., Saka, E., Badia, A., y Germain, R. (2008). A web usage mining framework for mining evolving user profiles in dynamic web sites. *Knowledge and Data Engineering, IEEE Transactions on*, 20(2):202–215.

Navigli, R. (2009). Word sense disambiguation: A survey. *ACM Computing Surveys (CSUR)*, 41(2):10.

Neville, J. y Jensen, D. (2003). Collective classification with relational dependency networks. En *Proceedings of the Second International Workshop on Multi-Relational Data Mining*, páginas 77–91.

Neyman, J. (1938). Contribution to the theory of sampling human populations. *Journal of the American Statistical Association*, 33(201):101–116.

Oard, D. W. (1997). The state of the art in text filtering. *User modeling and user-adapted interaction*, 7(3):141–178.

O'Reilly, T. (2005). Web 2.0: compact definition. *Message posted to http://radar.oreilly.com/2005/10/web-20-compact-definition.html*.

O'Reilly, T. (2007). What is web 2.0: Design patterns and business models for the next generation of software. *Communications & strategies*, (1):17.

Osowski, S. y Garanty, K. (2007). Forecasting of the daily meteorological pollution using wavelets and support vector machine. *Engineering Applications of Artificial Intelligence*, 20(6):745–755.

Page, L., Brin, S., Motwani, R., y Winograd, T. (1999). The pagerank citation ranking: bringing order to the web.

Pai, P.-F. y Lin, C.-S. (2005). Using support vector machines to forecast the production values of the machinery industry in taiwan. *The International Journal of Advanced Manufacturing Technology*, 27(1):205–210.

Pastor-López, I., Santos, I., Santamaria-Ibirika, A., Salazar, M., de-laPeña-Sordo, J., y Bringas, P. G. (2012). Machine-learning-based surface defect detection

and categorisation in high-precision foundry. En *Industrial Electronics and Applications (ICIEA), 2012 7th IEEE Conference on*, páginas 1359–1364. IEEE.

Pearl, J. (1982). *Reverend Bayes on inference engines: A distributed hierarchical approach*. Cognitive Systems Laboratory, School of Engineering and Applied Science, University of California, Los Angeles.

Peralta, J., Gutierrez, G., y Sanchis, A. (2008). Adann: automatic design of artificial neural networks. En *Proceedings of the 2008 GECCO conference companion on Genetic and evolutionary computation*, páginas 1863–1870. ACM.

Pielke, R. A. (2002). *Mesoscale meteorological modeling*. Academic press.

Pollock, D. S. G., Green, R. C., y Nguyen, T. (1999). *Handbook of time series analysis, signal processing, and dynamics*, volumen 1. Academic Press.

Postmes, T., Spears, R., y Lea, M. (1998). Breaching or building social boundaries? side-effects of computer-mediated communication. *Communication research*, 25(6):689–715.

Powell, M. J. (1987). Radial basis functions for multivariable interpolation: a review. En *Algorithms for approximation*, páginas 143–167. Clarendon Press.

Pyle, D. (1999). *Data preparation for data mining*, volumen 1. Morgan Kaufmann.

Quinlan, J. (1986). Induction of decision trees. *Machine learning*, 1(1):81–106.

Quinlan, J. (1993). *C4. 5 programs for machine learning*. Morgan Kaufmann Publishers.

Rahman, M. M., Bhattacharya, P., y Desai, B. C. (2007). A framework for medical image retrieval using machine learning and statistical similarity matching techniques with relevance feedback. *Information Technology in Biomedicine, IEEE Transactions on*, 11(1):58–69.

Rasmussen, C. E. (2006). Gaussian processes for machine learning.

Reed, R. D. y Marks, R. J. (1998). *Neural smithing: supervised learning in feedforward artificial neural networks*. Mit Press.

Resnick, P. y Miller, J. (1996). PICS: Internet access controls without censorship. *Communications of the ACM*, 39(10):87–93.

Resnick, P. y Zeckhauser, R. (2002). Trust among strangers in internet transactions: Empirical analysis of ebay's reputation system.

Resnick, P., Zeckhauser, R., Swanson, J., y Lockwood, K. (2006). The value of reputation on ebay: A controlled experiment. *Experimental Economics*, 9(2):79–101.

Robertson, A. M. y Willett, P. (1998). Applications of-grams in textual information systems. *Journal of Documentation*, 54(1):48–67.

Salton, G. y McGill, M. (1983). *Introduction to modern information retrieval*. McGraw-Hill New York.

Salton, G., Wong, A., y Yang, C.-S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620.

Santos, I., Laorden, C., y Bringas, P. G. (2011). Collective classification for unknown malware detection. En *Proceedings of the 6th International Conference on Security and Cryptography (SECRYPT)*, páginas 251–256.

Sapankevych, N. y Sankar, R. (2009). Time series prediction using support vector machines: a survey. *Computational Intelligence Magazine, IEEE*, 4(2):24–38.

Schneider, J., Kortuem, G., Jager, J., Fickas, S., y Segall, Z. (2000). Disseminating trust information in wearable communities. *Personal Technologies*, 4(4):245–248.

Schölkopf, B. y Smola, A. J. (2002). *Learning with kernels: support vector machines, regularization, optimization and beyond*. the MIT Press.

Schwartz, M. (2008). The trolls among us. *The New York Times Magazine*.

Sebastiani, F. (2006). Classification of text, automatic. *The Encyclopedia of Language and Linguistics*, 14:457–462.

Shachaf, P. y Hara, N. (2010). Beyond vandalism: Wikipedia trolls. *Journal of Information Science*, 36(3):357–370.

Shannon, C. (1951). Prediction and entropy of printed english. *Bell System Technical Journal*, 30(1):50–64.

Shiskin, J. y Eisenpress, H. (1957). Seasonal adjustments by electronic computer methods. *Journal of the American Statistical Association*, 52(280):415–449.

Singh, Y., Kaur, A., y Malhotra, R. (2009). Comparative analysis of regression and machine learning methods for predicting fault proneness models. *International Journal of Computer Applications in Technology*, 35(2):183–193.

Slutzky, E. (1937). The summation of random causes as the source of cyclic processes. *Econometrica: Journal of the Econometric Society*, páginas 105–146.

Smola, A. J. y Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and computing*, 14(3):199–222.

Srinivasan, S. y Wang, D. (2006). A supervised learning approach to uncertainty decoding for robust speech recognition. En *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, volumen 1, páginas I–I. IEEE.

Stauffer, D. R. y Seaman, N. L. (1990). Use of four-dimensional data assimilation in a limited-area mesoscale model. part i: Experiments with synoptic-scale data. *Monthly Weather Review*, 118(6):1250–1277.

Stevens, W. R. y Wright, G. R. (1994). *TCP/IP Illustrated: the protocols*, volumen 1. Addison-Wesley Professional.

Svetnik, V., Liaw, A., Tong, C., y Wang, T. (2004). Application of breiman's random forest to modeling structure-activity relationships of pharmaceutical molecules. *Multiple Classifier Systems*, páginas 334–343.

Swanson, N. R. y White, H. (1997). A model selection approach to real-time macroeconomic forecasting using linear models and artificial neural networks. *Review of Economics and Statistics*, 79(4):540–550.

Tadelis, S. (2003). Firm reputation with hidden information. *Economic Theory*, 21(2):635–651.

Tata, S. y Patel, J. M. (2007). Estimating the Selectivity of tf-idf based Cosine Similarity Predicates. *ACM SIGMOD Record*, 36(2):75–80.

Tay, F. E. y Cao, L. (2001). Application of support vector machines in financial time series forecasting. *Omega*, 29(4):309–317.

Thissen, U., Van Brakel, R., De Weijer, A., Melssen, W., y Buydens, L. (2003). Using support vector machines for time series prediction. *Chemometrics and intelligent laboratory systems*, 69(1):35–49.

Tobin, J. (1959). On the predictive value of consumer intentions and attitudes. *The review of economics and statistics*, 41(1):1–11.

Torkkola, K. (2003). Feature extraction by non parametric mutual information maximization. *The Journal of Machine Learning Research*, 3:1415–1438.

Tsang, E., Yeung, D., y Wang, X. (2003). Offss: optimal fuzzy-valued feature subset selection. *Fuzzy Systems, IEEE Transactions on*, 11(2):202–213.

Tukey, J. W. (1961). Discussion, emphasizing the connection between analysis of variance and spectrum analysis. *Technometrics*, 3(2):191–219.

Tumasjan, A., Sprenger, T. O., Sandner, P. G., y Welpe, I. M. (2010). Predicting elections with twitter: What 140 characters reveal about political sentiment. En *Proceedings of the fourth international aaai conference on weblogs and social media*, páginas 178–185.

Turner, T., Smith, M., Fisher, D., y Welser, H. (2005). Picturing Usenet: Mapping computer-mediated collective action. *Journal of Computer-Mediated Communication*, 10(4):00–00.

Üstün, B., Melssen, W., y Buydens, L. (2007). Visualisation and interpretation of support vector regression models. *Analytica chimica acta*, 595(1-2):299–309.

Vapnik, V. (1999). *The nature of statistical learning theory*. Springer.

Vapnik, V., Golowich, S. E., y Smola, A. (1997). Support vector method for function approximation, regression estimation, and signal processing. *Advances in neural information processing systems*, páginas 281–287.

Vasudevan, A. (2007). Wildcat: an integrated stealth environment for dynamic malware analysis.

Verhulst, P.-F. (1845). Recherches mathématiques sur la loi d'accroissement de la population. *Nouveaux mémoires de l'académie royale des sciences et belles-lettres de Bruxelles*, 18(2):3–38.

Wan, V. y Campbell, W. M. (2000). Support vector machines for speaker verification and identification. En *Neural Networks for Signal Processing X, 2000. Proceedings of the 2000 IEEE Signal Processing Society Workshop*, volumen 2, páginas 775–784. IEEE.

Whitby, A., Jøsang, A., y Indulska, J. (2005). Filtering out unfair ratings in bayesian reputation systems.

Wilbur, W. y Sirotkin, K. (1992). The automatic identification of stop words. *Journal of information science*, 18(1):45–55.

Wilks, Y. (1997). Information extraction as a core language technology. *Information Extraction A Multidisciplinary Approach to an Emerging Information Technology*, páginas 1–9.

Winters, P. R. (1960). Forecasting sales by exponentially weighted moving averages. *Management Science*, 6(3):324–342.

Wittel, G. L. y Wu, S. F. (2004). On attacking statistical spam filters. En *Proceedings of the first conference on email and anti-spam (CEAS)*. California, USA.

Ypma, A. y Heskes, T. (2003). Automatic categorization of web pages and user clustering with mixtures of hidden markov models. *WEBKDD 2002-Mining Web Data for Discovering Usage Patterns and Profiles*, páginas 35–49.

Yule, G. U. (1927). On a method of investigating periodicities in disturbed series, with special reference to wolfer's sunspot numbers. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 226:267–298.

Zhang, G., Eddy Patuwo, B., y Y Hu, M. (1998). Forecasting with artificial neural networks: The state of the art. *International journal of forecasting*, 14(1):35–62.

Zhang, G. P. (2003). Time series forecasting using a hybrid arima and neural network model. *Neurocomputing*, 50:159–175.

Zhang, Y., Jansen, B. J., y Spink, A. (2009). Time series analysis of a web search engine transaction log. *Information Processing & Management*, 45(2):230–245.

Zhou, Y., Jorgensen, Z., y Inge, M. (2007). Combating good word attacks on statistical spam filters with multiple instance learning. En *Tools with Artificial Intelligence, 2007. ICTAI 2007. 19th IEEE International Conference on*, volumen 2, páginas 298–305. IEEE.

Ziegler, C.-N. y Lausen, G. (2004). Spreading activation models for trust propagation. En *e-Technology, e-Commerce and e-Service, 2004. EEE'04. 2004 IEEE International Conference on*, páginas 83–97. IEEE.

Esta tesis doctoral se terminó de escribir en
Bérgamo (Italia) el 3 de Junio de 2014.