Universidad de las Ciencias Informáticas

Facultad 3

Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas

Método para la detección de información relevante de los comentarios de usuarios de aplicaciones de software en la UCI.

AUTORES:

Felix Javier Fonseca Torres

Gabriel Apolinaire Aguila

TUTORES:

Ing. Vladimir Milián Núñez

Lic. Raynel Batista Tellez

La Habana, 2017

Auspicio: Proyecto de investigación “Herramientas para el análisis de influencias en redes de innovación tecnológica”

"Tienes que empezar por la experiencia del cliente y luego ir hacia atrás, a la tecnología, y no al revés."

Steve Jobs

"El verdadero viaje del descubrimiento no consiste en buscar nuevos paisajes, sino en tener nuevos ojos."

Marcel Proust

DECLARACIÓN DE AUTORÍA

MINISTERIO DE EDUCACIÓN SUPERIOR

UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

Declaramos ser autores del presente trabajo de diploma y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales del mismo, con carácter exclusivo. Autorizamos a dicho centro para que haga el uso que estime pertinente con este trabajo.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año ____.

Felix Javier Fonseca Torres

Gabriel Apolinaire Aguila

Raynel Batista Téllez

Vladimir Milián Núñez

De Felix:

Agradezco a mi mamá, la persona que más quiero en el mundo, por ser incondicional y estar junto a mí en cada momento de mi vida dándome su apoyo y su cariño infinito.

A mi papá, por ser mi guía, mi consejero a la hora de tomar decisiones difíciles, por brindarme su apoyo en todo momento, por enseñarme a través de sus experiencias y sobre todo por ser mi amigo.

A mi hermana Daniela, mi querida hermana, la persona más especial y autentica que conocido, por ser la fuerza que me impulsa a alcanzar mis objetivos para servirle de guía e inspiración, por quererme tanto y siempre decirme la verdad por muy dura que sea.

A mi pequeña hermana Kamila, por adorarme y quererme tanto, por darme ese cariño tan sincero que solo ella sabe dar.

A mi amigo Aramis que ha sido mi apoyo y el de mi mamá durante mucho tiempo, que me ha tratado siempre como su hijo y siempre ha estado ahí para mí cuando lo he necesitado.

A mis abuelos, todos mis abuelos por ese cariño tan especial que solo una persona que ha vivido tanto sabe transmitir , por enseñarme la importancia de la unión familiar por la que siempre luchan, a mi abuelo fofito por darme sus consejos e inculcarme sus principios y preceptos de vida , a mi abuelita del alma mi viejita Elba que ha sido todo amor y cariño siempre conmigo, a mima por preocuparse siempre por mí y tenerme presente en todo momento, a papi que aunque ya no está me dio consejos a su forma los cuales me han servido de mucho, mis abuelos Carito y Aramis por su apoyo y cariño.

A mis tíos, todos mis tíos, mi tío Randy ese tío especial que se preocupa tanto por todos, a mi tía Zeida y mi tío Juan por ser tan atentos y preocupados conmigo, a mi tío Rubén mi policía siempre pendiente de todo lo que hago, mi tía Marle mi tía y mi tío Tony mis tíos locos y cariñosos; y en especial a mi tía Cari para la cual yo era su sol como ella solía decir.

A yermita por su cariño y apoyo, por regalarme a mi hermana Kamila.

A mis primos, Elisa, Laura, Alex y Aleannys, a los cuales he tomado como guía y ejemplo en los estudios y en la vida.

A todos mis familiares en general por su apoyo y cariño.

A mi familia adoptiva de Vegas, a mami Esther y a pipi por ser como mis padres, a mis padrinos Ylle y tata Fredy por su cariño y apoyo, a mis ahijados Dayneris, Disniel y Keila por su cariño.

A todos mis amigos y compañeros en especial a Román y a José con los que he compartido momentos malos y buenos, demostrándome siempre que puedo contar con ellos.

A mis tutores y a mi compañero de tesis, el Gabo, por la guía y el compañerismo durante este largo proceso.

En fin, agradezco a todos los que aportaron su granito de arena para que yo alcanzara este objetivo: a profesores, compañeros, amigos y familia.

De Gabriel:

A mi compañero de tesis, el Felix: por todo.

A nuestros tutores Vladimir y Raynel: por ser consejeros en la realización de este proyecto. Por amigos en el ámbito personal.

Gracias a Google, y a la comunidad de Stack Overflow.

A todos mis profesores: por haber contribuido con mi formación integral.

A todos mis compañeros de aula: desde el círculo infantil, primaria y secundaria. A los del IPVCE y la UCI, por convertir esas circunstancias en un hogar; de todos aprendí algo, y estoy agradecido por ello.

A los amigos de la infancia: por esa maravillosa estación llena de aventuras.

A todos los amigos con los que conviví: por compartir más que historias.

A los friky-amigos del café (o del PCP), a los que están y los que no: por los momentos buenos, y por los mejores.

A los amigos del malecón sin agua (de Santa Clara): por ser refugio a donde volver siempre (mi zona de confort).

Al Mejunje de Santa Clara: por ser multiverso y dimensión (ya no tan) desconocida. Por inspirarme y permitir escapar y encontrarme a través de sus historias, sus canciones, su gente…

A los amigos que están lejos (físicamente).

A toda mi **FAMILIA**:

Al batallón que me cuidó cuando nací. A los que viven lejos. A los que me vieron crecer. A los que se fueron sumando. A los que ya no están. Estoy eternamente agradecido.

A las damas que cargan **el trabajo más difícil del mundo,** mis madres: **Yamilet** y **Yaikel**.

A mi abuela querida: a **Mami**.

A mis hermanos: **Daniel** y **Yoanna**.

A mis padres: **Sergio**, **Jorge** y **Ariel**.

Estoy agradecido de la suerte que he tenido en la vida al haber coincidido con tantas personas especiales.

Muchas gracias a todos,

G.

RESUMEN

Los productos informáticos deben proporcionar una mejora en el desenvolvimiento, reducción de errores y ahorro de costos operativos para las empresas, debido a que el mercado de desarrollo de software es un entorno muy competitivo. Durante todo el proceso de desarrollo de la aplicación es necesario asegurar la calidad y robustez del software, ya sea a través de pruebas, encuestas o análisis de opiniones de usuarios, permitiendo a especialistas y a empresas de desarrollo de software evolucionar sus productos. El objetivo de esta investigación es desarrollar un método que permita la obtención de información relevante de los comentarios de usuarios de aplicaciones de software en la UCI. La solución obtenida se desarrolló sobre el marco de trabajo Django 1.6, el lenguaje de programación Python 2.7 y con el uso de las bibliotecas NLTK, sklearn y Gensim. El método presentado posee un alto nivel de rendimiento y efectividad, y contribuye a optimizar el proceso de obtención de información relevante de los comentarios de usuarios de software. La aplicación de mediciones basadas en el coeficiente de correlación de Spearman y Pearson verificó la correspondencia entre los datos obtenidos en los experimentos y aquellos obtenidos de la solución implementada. Las pruebas realizadas han corroborado que al utilizar el complemento implementado se minimiza el esfuerzo dedicado y se elimina el efecto de la influencia de la subjetividad al analizar los comentarios de forma tradicional, demostrando su utilidad y placer de uso.

Palabras Clave: agrupamiento, comentarios de usuarios, minería de texto, procesamiento del lenguaje natural.

ÍNDICE

ÍNDICE DE FIGURAS

INTRODUCCIÓN

Los productos informáticos deben proporcionar una mejora en el desenvolvimiento, reducción de errores y ahorro de costos operativos para las empresas, debido a que el mercado de desarrollo de software es un entorno muy competitivo. Es por ello que el desarrollo de una aplicación de software no termina con su liberación o disponibilidad a los clientes, pues las empresas y desarrolladores necesitan mejorar sus aplicaciones con el fin de aumentar las ventas e ingresos.

Durante todo el proceso de desarrollo de la aplicación es necesario asegurar la calidad y robustez del software, ya sea a través de pruebas, encuestas o análisis de opiniones de usuarios. La realización de pruebas de software se utiliza para: detectar fallas o posibles errores en el sistema y para comprobar su rendimiento. A pesar de que ayudan al mejoramiento del sistema que se desarrolla, nunca alcanzan a exigir a una aplicación, como pueden hacerlo cientos de usuarios usándola en diferentes ambientes y en un grupo heterogéneo de dispositivos.

Además de las pruebas masivas de software, existe la retroalimentación de información sobre el uso de las aplicaciones informáticas por los usuarios mediante la realización de encuestas o pruebas de aceptación. Las encuestas son elaboradas por un grupo de expertos, asegurándose de que una muestra representativa de las personas conteste la misma con el objetivo de obtener conocimientos valiosos con el fin de maximizar el éxito. El esfuerzo detrás de este proceso no es despreciable y en muchas ocasiones no es factible, pues los usuarios no comprenden la necesidad de la veracidad de los datos con que llenan estas interrogantes.

Una tercera posibilidad es aprovechar una valiosa fuente de información: las revisiones de los usuarios. Existen varios portales en los cuales los usuarios de una aplicación específica pueden dejar un comentario, mediante la asignación de una puntuación (generalmente expresado con un número de estrellas que van de uno a cinco). Además, cuentan con un campo de texto libre para informar errores, recomendar nuevas características, o simplemente describir sus sentimientos y experiencias durante el uso de la aplicación.

Estas opiniones, escritas en un lenguaje coloquial y natural, representan una fuente valiosa de información para los creadores de software. Su análisis permite obtener recomendaciones sobre cómo mejorar la aplicación y entender lo que necesitan los usuarios (Allic, 2010). Esta información puede ser relevante para los desarrolladores cuando se desea liberar una nueva versión de su aplicación o para estudiar la competencia.

Algunas aplicaciones, como Google Play [1] proporcionan apoyo a los desarrolladores con la posibilidad de clasificar las revisiones por calificación para detectar fácilmente los más críticos. Sin embargo, algunos usuarios simplemente asignan puntuaciones muy bajas (por ejemplo, una estrella) sólo para aumentar la visibilidad de su revisión y en muchos casos es más positivo que negativo su contenido. Esto reduce claramente la visibilidad de otros comentarios que podrían contener información sobre errores, ocultándolos de los ojos de los desarrolladores. Por lo tanto, este apoyo trivial no es suficiente.

Con el fin de obtener ventajas de manera efectiva de la información que los usuarios dejan en las opiniones, un desarrollador tiene que navegar a través de todo el conjunto de comentarios, leer manualmente cada revisión y comprobar si contiene información relevante (por ejemplo, sugerencias para nuevas características o de mejoras). Este proceso no es factible si la aplicación recibe cientos de comentarios por día, como sucede con las aplicaciones populares.

Por otra parte, alrededor del 70% de las opiniones de los usuarios no resultan relevantes para los desarrolladores porque muchas se repiten, son similares o simplemente son comentarios neutros sobre el uso de la aplicación (Jojo, 2011). Dicha situación conlleva a que los desarrolladores deben distinguir entre la información relevante y no relevante, invirtiendo tiempo en el análisis de esta información y alargando el tiempo de liberación de la(s) versión(es) planificada(s).

La Universidad de las Ciencias Informáticas (UCI), con más de 200 proyectos y autora del 76% de las soluciones implantadas en el país y del 99% de las soluciones informáticas exportadas por la nación (Universidad de las Ciencias Informáticas, 2016), no ha definido una infraestructura encaminada a mejorar el diseño de experiencia de usuario de sus productos. Al no tener presente las opiniones de los usuarios pueden liberarse nuevas versiones de la aplicación con características no deseadas o con errores, lo que resultaría una posible pérdida de nuevos o antiguos usuarios, e incluso la correspondiente disminución monetaria.

Partiendo de la situación problemática expuesta anteriormente surge el siguiente **problema a resolver:** El modo en que se **procesan los comentarios de usuarios** en la UCI no contribuye a **precisar la información relevante** para la evolución de las aplicaciones.

[1] Tienda de aplicaciones Android, disponible en: https://play.google.com/

Se delimita como **objeto de estudio**: minería de texto, específicamente en el **campo de acción**: análisis de comentarios de usuarios.

Para dar solución al problema a resolver se define como **objetivo general**: Desarrollar un método de agrupamiento de comentarios de usuarios que permita la detección de información relevante de los comentarios de usuarios de aplicaciones de software en la UCI.

Para dar cumplimiento al objetivo general se definen los siguientes **objetivos específicos:**

- Analizar los enfoques utilizados por los algoritmos de agrupamiento en la minería de texto para el agrupamiento de comentarios.
- Diseñar un método que posibilite el agrupamiento de texto por temáticas a partir de los comentarios de usuarios.
- Implementar un complemento a partir del método diseñado para el agrupamiento de texto por temáticas partir de los comentarios de usuarios.
- Evaluar el comportamiento del método implementado a partir de su utilización en un entorno real.

Idea a defender: si se desarrolla un **método de minería de texto que agrupe los comentarios de usuarios por temáticas**, entonces, se elevará la **precisión con que se obtiene la información relevante al analizar los comentarios de usuarios** de aplicaciones de software desarrolladas en la UCI.

Las **tareas de investigación** que se deben cumplir son**:**

- Análisis de los principales conceptos y trabajos relacionados con el análisis de comentarios en la minería de texto.
- Análisis de los algoritmos existentes para el agrupamiento de comentarios en la minería de texto.
- Diseño de un método para el agrupamiento de texto por temáticas a partir de los comentarios de usuarios.
- Implementación del método diseñado permitiendo su integración a la herramienta AOpinion.
- Generación de los datos necesarios para validar la propuesta.
- Aplicación de las pruebas necesarias y análisis de los resultados obtenidos en la validación.

Se espera obtener un método que sea capaz de agrupar los comentarios de los usuarios por temáticas para facilitar el trabajo con las revisiones de usuarios a los desarrolladores de software. Los comentarios de usuarios constituyen una ayuda para que los integrantes de los equipos de desarrollo conozcan las opiniones de los clientes, relacionados con las funcionalidades del software, permitiéndoles mejorar el producto y erradicar no conformidades.

Para el desarrollo del presente trabajo se emplean los siguientes **métodos científicos**:

Analítico-sintético: permitió analizar individualmente los principales conceptos relacionados con el área de estudio a tratar, posibilitando un análisis profundo de cada uno, para luego llevar a cabo el estudio de las relaciones que se establecen entre ellos.

Hipotético-deductivo: se utilizó para guiar la investigación desde el planteamiento del problema hasta la verificación de la solución a partir de las validaciones, orientando la secuencia lógica de las tareas que se realizaron.

Histórico-Lógico: se empleará para estudiar la trayectoria y el desarrollo histórico de las herramientas que utilizan el análisis de comentarios de usuarios y poder comprender la lógica de sus aportes, así como las tendencias actuales.

Experimentación: se empleó para verificar la correspondencia entre los resultados obtenidos con la aplicación del método propuesto y los resultados de la aplicación de técnicas tradicionales de análisis de comentarios.

El presente documento está estructurado en tres capítulos cuyos contenidos son:

- **Capítulo 1:** se describen los conceptos asociados al objeto de estudio: el proceso de descubrimiento de conocimiento (KDD) y la minería de texto. Se detallan además las técnicas de la minería de texto. Se realiza un estudio de las herramientas que emplean el análisis de comentarios de usuarios. Se define el lenguaje de programación a utilizar y las herramientas necesarias para realizar el presente trabajo de diploma.
- **Capítulo 2:** se describe brevemente cuáles son los pasos a seguir para desarrollar el proceso de solución propuesto, así como las métricas y algoritmos a que se utilizan. La metodología adoptada en este trabajo está basada en las etapas que caracterizan el modelo de proceso CRISP-DM: comprensión del ámbito de aplicación, comprensión de los datos, preparación de los datos, modelado, evaluación y desarrollo.
- **Capítulo 3:** se realiza la validación de la solución propuesta. Se mide la precisión del algoritmo de agrupamiento a partir de las métricas de evaluación de rendimiento, que comprueban la eficiencia con que se ejecutan y la calidad de las soluciones que éstos

producen. Se comprueba la utilidad del método al aplicar dos experimentos que prueban, que al utilizar el complemento implementado se minimiza el esfuerzo dedicado, y se elimina el efecto de la influencia de la subjetividad al analizar los comentarios de forma tradicional. Además se confronta el placer de uso entre realizar las tareas de manera tradicional y usando el complemento implementado.

CAPÍTULO 1. Fundamentación teórica

En este capítulo se realiza un análisis general del proceso de descubrimiento de conocimiento en bases de datos (en lo adelante KDD), así como de las técnicas utilizadas en la minería de texto. Se describen de forma resumida las principales características, conceptos y aplicaciones de cada una, orientadas fundamentalmente al dominio de los comentarios de usuarios.

1.1 Análisis de comentarios de usuarios. Análisis conceptual

Los comentarios de usuarios son una herramienta de gran ayuda para estudiar lo que desean y piensan los mismos en los lanzamientos de productos de software, y así, contribuir a desarrollar aplicaciones de mejor calidad con menor tiempo de respuesta. Brindarles a los usuarios la posibilidad de emitir sus opiniones, permite procesar y extraer información que es realmente importante, descubriendo conocimiento y patrones en estos datos (Minería de Datos como soporte a la toma de decisiones empresarial, 2007).

Estos autores abordan con gran precisión el significado de considerar los comentarios de usuarios como expresiones escritas en lenguaje natural, que expresan experiencias adquiridas por los clientes en la interacción con un software determinado. Sus ideas constituyen herramientas de gran ayuda cuando se desea conocer el criterio de los clientes en los lanzamientos de software. A continuación, se describe la investigación en el ámbito internacional y nacional, respecto a sistemas relacionados con el campo de acción predeterminado, obteniendo los resultados que se muestran en el siguiente epígrafe.

1.1.1 Investigación de las herramientas de análisis de comentarios

Google PLAY[2] (tienda oficial de Google para terminales Android, distribuye películas, música, libros y, sobre todo, aplicaciones) sirve de apoyo a los desarrolladores, pues posibilita la clasificación de las revisiones por calificación para detectar fácilmente los más críticos. Sin embargo, algunos usuarios asignan puntuaciones muy bajas sólo para aumentar la visibilidad de su revisión y en muchos casos es más positiva que negativa su contenido (Crider, 2014). Esto reduce claramente la visibilidad de otras críticas que podrían contener información valiosa acerca de los errores, mal funcionamiento o mejoras, ocultándolos de los ojos de los desarrolladores. Por lo tanto, este apoyo

[2] Tienda de aplicaciones de Android, disponible en: https://play.google.com/store

no es suficiente. Otras aplicaciones parecidas son **App Store**[3](servicio para que los usuarios busquen y descarguen sitios informáticos) y **AndroidPit** [4] (permite realizar búsquedas de aplicaciones, ofrece numerosas recomendaciones propias).

Brandwatch es una herramienta de monitoreo de redes sociales que permite obtener una idea de la opinión pública general sobre ciertos temas. Utiliza un proceso de reglas para ayudar a entender mejor las diferentes maneras en que el contexto puede afectar al sentimiento. Es una herramienta abarcadora, el precio más básico es de 800 dólares y en la medición de 'sentimiento' no es posible conocer en más detalle al usuario que emitió el comentario negativo o positivo (Mendoza, Milagros, 2014).

Otras aplicaciones semejantes son **HootSuite** (herramienta gratuita disponible para la gestión de las redes sociales, ya que cubre múltiples plataformas tales como: Twitter, Facebook, LinkedIn, WordPress, Foursquare y Google+), **Social Mention** (una de las mejores herramientas gratuitas de monitorización disponible en el mercado, pues analiza la información de una manera más profunda y también mide la influencia en cuatro categorías: *Strenght* (Fortaleza), *Sentiment* (Sentimiento), *Passion* (Pasión) y *Reach* (Alcance).

ITelligent, es un sistema de minería de opinión para inteligencia comercial; **OPAL**, complemento de Drupal que analiza comentarios realizados por los usuarios y detecta si el resultado es positivo, negativo o neutral; **IIC-Lynguoque**, es un conjunto de herramientas que ayudan a extraer la opinión positiva, negativa y neutra de un texto; **netOpinion**, permite conocer opiniones de productos o servicios en foros y redes sociales; **WebOpinion**, gestiona mes a mes la evolución de la imagen de un usuario en la red; y **Sentitex,** que consiste en un conjunto de aplicaciones para el análisis de sentimientos en textos. Desafortunadamente la mayoría de estas herramientas son propietarias y es necesario pagar la licencia para su utilización, ya que son concebidas con fines comerciales. Adicionalmente, cada una de ellas responde a objetivos específicos de la minería de opinión (PosNeg opinion: Una herramienta para gestionar comentarios de la web., 2015).

PosNeg Opinion es una herramienta desarrollada por el Departamento de Ciencia de la Computación en la Universidad Central de las Villas, para que el usuario analice un gran cúmulo de opiniones de manera sencilla, ya que se convierten los ficheros XML a texto plano. Fue desarrollada

[3] Tienda de aplicaciones de IOS, disponible en: http://www.app-store.es/

[4]Tienda de aplicaciones de Android, disponible: en http://www.androidpit.es/aplicaciones-android

completamente en JAVA, por lo que es multiplataforma, necesita como entrada un fichero XML con todas las opiniones a analizar y como salida muestra cuántas son positivas y cuántas negativas. A petición del usuario también retorna el porcentaje, además de una lista con las opiniones negativas y positivas (PosNeg opinion: Una herramienta para gestionar comentarios de la web., 2015).

Una vez descritas las herramientas antes expuestas se puede concluir que varias son propietarias y resulta necesario pagar la licencia o altos precios para su utilización, debido a que fueron concebidas con fines comerciales. Las herramientas de monitoreo de redes sociales como **Brandwatch** son abarcadoras y algunas de ellas no controlan los artículos de noticias generales o foros. **PosNeg Opinion** es una aplicación de escritorio para devolver las opiniones negativas y positivas, sin tener en cuenta clasificaciones o grupos específicos.

Las herramientas estudiadas realizan análisis de comentarios de usuarios según sus necesidades, pero no satisfacen la solución del problema de la investigación, por lo que desarrollar una herramienta propia es necesario.

Por consiguiente, el objetivo del trabajo es desarrollar un método que permita agrupar los comentarios de usuario de manera no supervisada, que muestre al usuario información relevante y que sea agregado como un complemento al sistema **AOpinion** para agrupar los comentarios de los usuarios subidos de un archivo .txt.

1.2 Proceso de descubrimiento de conocimiento (KDD)

El proceso de descubrimiento de conocimiento (KDD, por sus siglas del inglés *Knowledge Discovery in Databases*) se define como el proceso no trivial de identificar patrones con cierto grado de incertidumbre, que aporten información desconocida y útil tanto para el sistema, como para el usuario, y en última instancia comprensibles a partir de los datos (“…”) se trata de un proceso complejo que tiene como objetivo descubrir conocimiento útil (Hernández Orallo, y otros, 2004). El KDD está organizado entorno a cinco fases (Figura 1):

1. **Integración y recopilación de datos**: se determinan las fuentes de información que pueden ser útiles, seguidamente se transforman los datos a un formato común, unificando toda la información recogida, detectando y resolviendo inconsistencias.
2. **Selección, limpieza y transformación:** en esta fase se eliminan o corrigen los datos incorrectos y se decide la estrategia a seguir con los datos incompletos.
3. **Minería de datos**: se decide cuál es la tarea a realizar y se elige el método que se va a utilizar.
4. **Evaluación e interpretación:** se evalúan los patrones y se analizan los expertos.

5. **Difusión y uso:** en esta última fase se hace uso del nuevo conocimiento y se hace partícipe de él a todos los posibles usuarios.

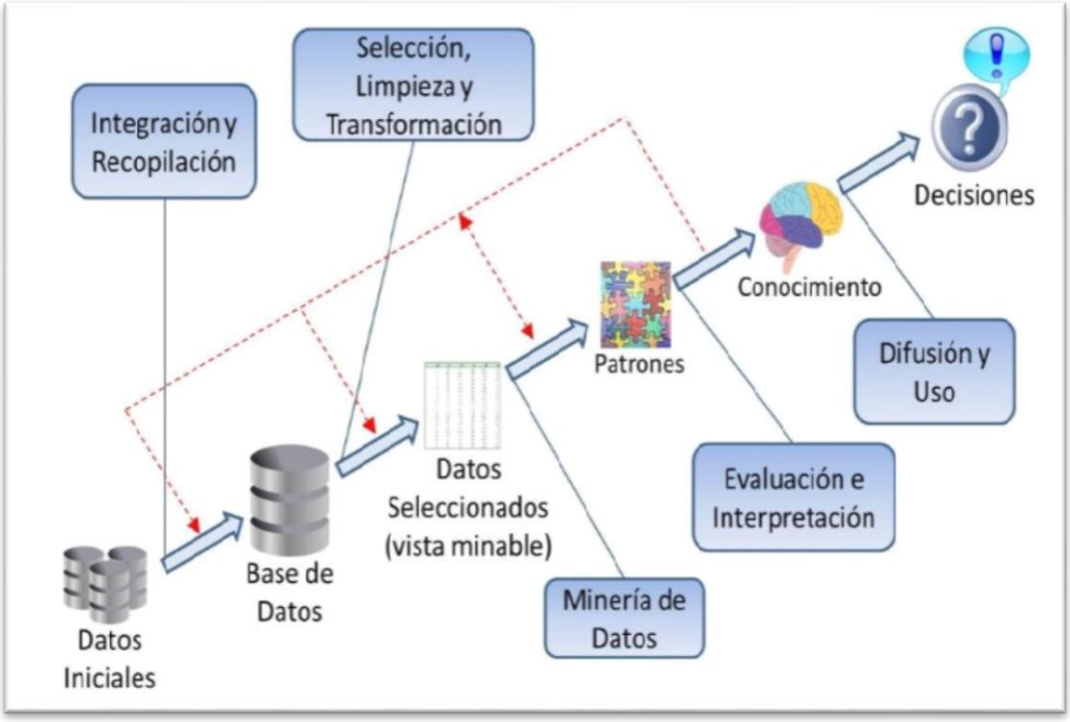

Figura 1: Proceso KDD (Hernández Orallo, y otros, 2004)

1.3 Minería de texto

Tradicionalmente la búsqueda de conocimiento se ha realizado sobre datos almacenados en bases de datos, pero la mayoría de la información dentro de una organización se encuentra en formato de texto, ya sea Intranet, páginas Web, informes de trabajo, publicaciones, correos electrónicos, entre otros. De esta manera, se abre un nuevo camino en la extracción de conocimiento de los documentos; de esta gran necesidad surge la técnica o proceso llamado: minería de texto. La minería de texto difiere de minería de datos en el trato de la información, donde la información textual difiere de la estructurada principalmente en la ausencia de estructura o en la compleja estructura implícita del texto. De este modo, se hace necesario buscar alguna representación intermedia del texto que pueda ayudar a la aplicación de técnicas de descubrimiento, que permita extraer patrones útiles (Larsen, y otros, 1999).

La minería de texto un proceso que consiste en extraer información útil de conjuntos de documentos no estructurados de texto, e identificar automáticamente patrones interesantes no triviales o conocimiento (Feldman, y otros, 1995) (Feldman, y otros, 2007).

Para poder descubrir conocimiento en texto, se debe pasar por algunas etapas importantes en este proceso, como es la etapa del pre-procesamiento que le da al texto una forma intermedia que permita ser tratada computacionalmente, luego aplicar alguna técnica de minería de texto y finalmente la visualización de los resultados.

Figura 2 Etapas de la Minería de Texto (Viera, 2017)

1.3.1 Procesamiento de lenguaje natural (NLP, por sus siglas del inglés Natural Language Proccessing)

El lenguaje natural se entiende como el lenguaje hablado y escrito con el propósito que exista comunicación entre una o varias personas, es más directo para expresar lo que se quiere comunicar (Benavides Cañón, y otros).

El procesamiento del lenguaje natural es una disciplina que relaciona directamente la computación y la lingüística, tiene como principal objetivo conseguir que el lenguaje humano pueda utilizarse como entrada en un proceso automatizado (Hernández, 2014).

Es un área de investigación en continuo desarrollo, se aplica en la actualidad en diferentes actividades como son la traducción automática, sistemas de recuperación de información, elaboración automática de resúmenes, e interfaces en lenguaje natural (Aplicaciones del procesamiento del lenguaje natural en la recuperación de información en español, 2005).

Pre-procesamiento básico del texto

El pre-procesamiento básico del texto, consiste en eliminar todos los caracteres no alfanuméricos del texto (como signos de puntuación, etc.), se convierten todas las palabras en minúsculas y se eliminan los acentos. El resultado de este paso es un texto en el que las palabras no contienen ningún carácter no alfanumérico, están en minúsculas, sin acentos, y separadas entre sí por un solo espacio.

Eliminar palabras de parada y realizar stemming (radicalizar)

Consiste en eliminar el "ruido" del texto de revisión de los usuarios. Este ruido se compone por las palabras de parada, que son las palabras más comunes en un idioma y las palabras de menos de

tres caracteres. También, se aplica stemming a las palabras con el fin de reducirlas a su raíz. El resultado de este paso es un texto que se caracteriza por los n-gramas extraídos que los componen, pero eliminando el ruido y aplicando stemming a los n-gramas (Pérez, 2015).

Modelo de espacios vectoriales (Matriz documento-término)

Muchas de las tareas de recuperación de información como la búsqueda, agrupamiento o categorización de textos tienen como primer objetivo procesar documentos en lenguaje natural. El problema que surge es que los algoritmos que pretenden resolver estas tareas necesitan representaciones internas explícitas de los documentos (Polettini, 2004).

En el área de recuperación de información normalmente se usa una expresión vectorial, donde las dimensiones del vector representan términos, frases o conceptos que aparecen en el documento. En este aspecto la representación más adoptada es la conocida como bolsa de palabras: una colección de documentos compuesta por *n* documentos indexados y *m* términos representados por una matriz documento-término de *n* x *m*. Donde los *n* vectores renglón representan los *n* documentos; y el valor asignado a cada componente refleja la importancia o frecuencia ponderada que produce el término, frase o concepto t_i en la representación semántica del documento d_j.

$$d_j = (w_{1j}, w_{2j} \dots w_{mj})$$

Donde *m* es la cardinalidad del diccionario y representa la contribución del término t_i para la representación semántica del documento d_j.

En esta representación vectorial de documentos, el éxito o fracaso se basa en la ponderación o peso de los términos. Aunque ha habido mucha investigación sobre técnicas de ponderación de términos, en realidad no hay un consenso sobre cuál método es el mejor (Polettini, 2004). También hay que destacar que el espacio de renglones de la matriz documento-término determinan el contenido semántico de la colección de documentos. Sin embargo, una combinación lineal de dos vectores-documento no representa necesariamente un documento viable de la colección. Más importante aún, mediante el modelo espacio vectorial se pueden explotar las relaciones geométricas entre dos vectores-documento (y términos) a fin de expresar las similitudes y diferencias entre términos (Xu, y otros, 2003).

Si bien el rendimiento de un sistema de recuperación de información depende en gran medida de las medidas de similitud entre documentos, la ponderación de términos desempeña un papel fundamental para que esa similitud entre documentos sea más confiable. Así, por ejemplo, mientras que una representación de documentos basada solo en las frecuencias o apariciones de términos no es capaz de representar adecuadamente el contenido semántico de los documentos (Berry, y

otros, 2005), la representación de términos ponderados hace frente a errores o incertidumbres asociadas a la representación simple de documentos.

Una colección de *n* documentos indexados por *m* términos puede ser representada por una matriz A de dimensión *n* x *m*, donde cada elemento a_{ij} es usualmente definido por una frecuencia ponderada del término *i* en el documento *j* cuyo objetivo principal es mejorar el rendimiento en la recuperación de información; entendiendo como rendimiento la habilidad de recuperar información relevante y descartar información irrelevante. La siguiente figura (Figura 3) muestra una matriz documento-término simple, donde cada columna representa un término en la colección, cada renglón un documento y cada celda o elemento de la matriz la ocurrencia del término en el documento.

	Término 1	Término 2	Término 3
Documento 1	**1**	**0**	**0**
Documento 2	**0**	**0**	**1**
Documento 3	**1**	**1**	**1**
Documento 4	**0**	**1**	**0**

Figura 3: Ejemplo de matriz documento-término (Guevara López, 2011)

En ella se observa que el término 1 aparece en el documento 1 y 3, pero no en los otros dos documentos. Se demuestra así que cada renglón de la matriz de 4×3 puede ser representado en un espacio de tres dimensiones (Guevara López, 2011).

1.3.2 Similitud entre documentos

Una medida natural para comparar dos documentos es la similitud. Dos documentos son similares cuando comparten las mismas palabras, así que mientras más palabras se compartan más similares son los documentos. Algo muy relevante de la medida de similitud es que las palabras que ocurren en un documento, pero no en los otros son ignoradas (lo que no ocurre en las medidas de distancia convencionales).

Antes del agrupamiento, se debe determinar una medida de similitud / distancia. La medida refleja el grado de cercanía o separación de los objetos objetivo y debe corresponder a las características

que se cree que distinguen los grupos incrustados en los datos. En muchos casos, estas características dependen de los datos o el contexto del problema en cuestión, y no hay ninguna medida que sea universalmente mejor para todo tipo de problemas de agrupación. En general, las medidas de similitud / distancia muestran la distancia o similitud entre la descripción simbólica de dos objetos en un único valor numérico, que depende de dos factores: las propiedades de los dos objetos y la medida misma (Huang, 2008).

Dada la diversidad de las medidas de similitud y distancia disponibles, su eficacia en la agrupación de documentos de texto aún no está clara. Con el fin de obtener buenos resultados se evalúan cinco medidas con experimentos empíricos: distancia euclidiana, similitud de coseno, coeficiente de Jaccard, coeficiente de correlación de Pearson y divergencia media de Kullback-Leibler.Por los resultados obtenidos en los experimentos se decidió utilizar la similitud del coseno para el algoritmo Ward y la distancia euclídea para el algoritmo K-means, ya que con el conjunto de datos utilizados para la evaluación arroja los mejores resultados (Huang, 2008).

Métricas

No todas las medidas de distancia son métricas. Para calificar como una métrica, una medida d debe satisfacer las siguientes cuatro condiciones.

Sea x e y dos objetos en un conjunto y d (x, y) la distancia entre x e y.

1. La distancia entre dos puntos cualquiera debe ser no negativa, Es decir, d (x, y)>= 0.
2. La distancia entre dos objetos debe ser cero si y solo si los dos objetos son idénticos, esto es, d (x, y) = 0 si y solo si x = y.
3. La distancia debe ser simétrica, es decir, la distancia de x a y es la misma que la distancia de y a x, es decir, d (x, y) = d (y, x).
4. La medida debe satisfacer la desigualdad triangular, que es d (x, z) =< D (x, y) + d (y, z).

Distancia euclídea

La distancia euclídea es una métrica estándar para problemas geométricos. Es la distancia ordinaria entre dos puntos y se puede medir fácilmente con una regla en el espacio de dos o tres dimensiones. La distancia euclídea es ampliamente utilizada en los problemas de agrupamiento, incluyendo el agrupamiento de texto. Satisface las cuatro condiciones anteriores por lo tanto es considerada una métrica. También es la medida de distancia por defecto usada con el algoritmo K-means (Huang, 2008).

La medición de la distancia entre documentos de texto, dado dos documentos d_a y d_b representados por sus vectores de término t_a y t_b respectivamente, la distancia euclidiana de los dos documentos se define como:

$$\mathrm{D}_E(\overrightarrow{t_a}, \overrightarrow{t_b}) = (\sum_{t=1}^{m} |w_{t,a}, w_{t,b}|^2)^{1/2}$$

Similitud coseno

Cuando los documentos se representan como vectores de término, la similitud de dos documentos corresponde a la correlación entre los vectores. Esto se cuantifica como el coseno del ángulo entre vectores, es decir, la semejanza de coseno. La similitud de coseno es una de las medidas de similitud más populares aplicadas a los documentos de texto, como en numerosas aplicaciones de recuperación de información y en algoritmos de agrupamiento (Huang, 2008).

Dado los dos documentos t_a y t_b, su similitud del coseno es:

$$SIM_C(\overrightarrow{t_a}, \overrightarrow{t_b}) = \frac{\overrightarrow{t_a} \cdot \overrightarrow{t_b}}{|\overrightarrow{t_a}| \times |\overrightarrow{t_b}|}$$

Donde t_a y t_b son vectores m-dimensionales sobre el set de términos T= {t_1, …, t_m}. Cada dimensión representa un término con su peso en el documento, que es no negativa. Como resultado, la similitud del coseno es no negativa y se encuentra entre 0 y 1.

Una propiedad importante de la similitud coseno es su independencia de la longitud del documento. Por ejemplo, al combinar dos copias idénticas de un documento d para obtener un nuevo pseudo documento d', la similitud de coseno entre d y d' es 1, lo que significa que estos dos documentos se consideran idénticos.

Mientras tanto, dado otro documento l, d y d' tendrá el mismo valor de similitud a l, es decir, sim (t_d, t_l) =sim ($t_{d'}$, t_l). En otras palabras, los documentos con la misma composición, pero diferentes totales son tratados de forma idéntica. Estrictamente hablando, esto no satisface la segunda condición de una métrica, porque después de todo la combinación de dos copias es un objeto diferente del documento original. Sin embargo, en la práctica, cuando los vectores término se normalizan a una longitud unitaria tal como 1, y en este caso la representación de d y d' es la misma.

1.3.3 Agrupamiento de documentos

El agrupamiento es una técnica de aprendizaje automático que tiene por objetivo agrupar un conjunto de objetos en subconjuntos o grupos. Un grupo de objetos tiene como característica que sus miembros son muy similares entre ellos y substancialmente distintos a los objetos de los otros

grupos. Todos los problemas de agrupamiento son, en esencia, problemas de optimización, en los que se maximiza o minimiza una función de calidad, generalmente en términos de similitud entre los objetos, sujeta a un conjunto de restricciones (Steinbach, 2000).

Un algoritmo de agrupamiento minimiza la distancia intra-grupo y maximiza la distancia inter-grupo, usando una apropiada medida de distancia entre los objetos, es decir, se agrupan los objetos similares y se separan aquellos que no lo sean.

A diferencia del problema de clasificación, en el problema de agrupamiento no hay clases predefinidas, dado que no se tiene la información a priori sobre la clase a la que pertenece un objeto, aunque cada grupo que se forma puede ser visto como una clase de objetos. El agrupamiento es una de las formas más comunes de aprendizaje no supervisado (Steinbach, 2000).

El análisis de grupos permite contribuir a la comprensión de la información cuando la cantidad de datos es muy grande y su procesamiento muy demandante, entregando una estructura adicional a la muestra de datos a través del descubrimiento de distribuciones y correlaciones entre los conjuntos de datos. También permite inferir algunas hipótesis referentes a los datos o validar hipótesis específicas.

Dentro de las categorías de agrupamiento más utilizados destacan dos tipos: agrupamiento particional y agrupamiento jerárquico. En el agrupamiento particional se intenta directamente descomponer el conjunto de objetos en un conjunto de grupos disjuntos, optimizando algún criterio predefinido o función objetivo en un procedimiento iterativo. En el agrupamiento jerárquico se procede sucesivamente juntando grupos de un miembro cada uno hasta construir un solo grupo con todos los objetos (método aglomerativo) o viceversa (método divisivo) y el resultado es un árbol de grupos (dendrograma) en el cual se muestra cómo los grupos están relacionados. Para el agrupamiento de grandes conjuntos de documentos de textos los algoritmos de agrupamiento particionales resultan más adecuados, debido a sus requerimientos computacionales relativamente bajos (Salinas Dezerega, 2016).

Considerando el constante incremento de contenido de textos (contenidos disponibles en internet, bibliotecas digitales e información personal digitalizada) la aplicación de agrupamientos de documentos resulta muy útil para tareas como: encontrar documentos similares, organizar y buscar eficientemente en grandes colecciones de documentos, detectar contenidos duplicados (detección de plagio), etiquetar colecciones de documentos, entre otros. Para obtener resultados útiles y eficientes a partir del agrupamiento de documentos, se debe realizar primeramente una apropiada selección de atributos, definir el tipo de medida de similitud y especificar el algoritmo de agrupamiento a utilizar.

En el agrupamiento de documentos se asigna cada documento a un grupo, agrupando los documentos similares dentro de un mismo grupo y separándolos de los otros grupos formados por documentos diferentes. En otras palabras, los documentos en un grupo comparten una temática y los documentos en diferentes grupos representan distintas temáticas. Para este tipo de agrupamiento, la medida de similitud asume un rol muy importante, dado que los documentos no poseen una clase predefinida por lo que la similitud refleja el grado de cercanía o separación de los objetos y define la asignación de clases.

Los grupos de documentos formados tienen un significado implícito en su agrupamiento, por lo que una correcta caracterización de un grupo mediante ciertas palabras claves puede entregar un significado útil para la interpretación de cada grupo. Los resultados de la aplicación del agrupamiento deben ser evaluados por algún método definido previamente e interpretados correctamente basados en alguna otra evidencia experimental.

Los mayores inconvenientes que puede tener el agrupamiento de documentos son la alta dimensionalidad del espacio de atributos que afecta su eficiencia y los atributos redundantes o irrelevantes que pueden alterar los resultados (Salinas Dezerega, 2016).

En esta investigación, se dispone cada comentario como si fuera un documento.

1.3.3.1 Algoritmo K-means

Este algoritmo particiona los N objetos en K particiones (K siendo un valor arbitrario) en donde un objeto va al grupo con la media más cercana. El algoritmo asigna K centros aleatoriamente, luego asigna los objetos al centro más cercano. El centro se recalcula como la media de los puntos que tiene asignado, una vez actualizado se vuelven a reasignar los objetos al más cercano y hasta tener convergencia (Blanco-Hermida Sanz, 2016).

Se define el centro de un grupo como aquel elemento que minimiza la suma de las similitudes al resto de los elementos del grupo:

$$mC = argMin \cdot m \in C \sum \cdot mj \in C \; dist(m, mj)$$

Este algoritmo depende mucho de la asignación inicial de los centros, y puede dar un resultado u otro por lo que es mejor hacer varias pruebas con diferentes valores. Una variante llamada K-means++ intenta resolver este problema al escoger mejores centros.

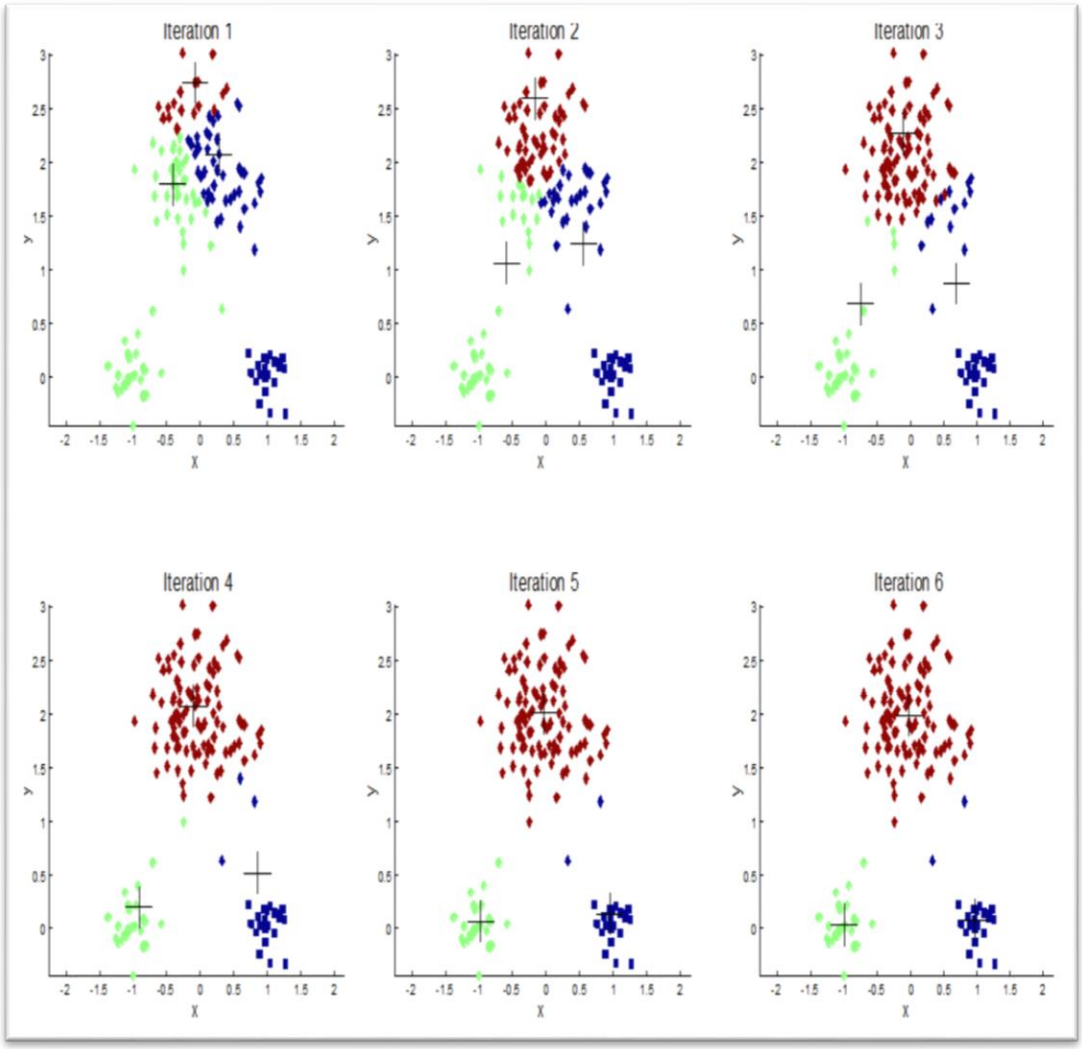

Figura 4: Iteraciones del algoritmo K-means (Blanco-Hermida Sanz, 2016)

En la primera iteración se pueden observar tres cruces correspondientes a los 3 centros iniciales. Se han asignado los puntos al centro más cercano, es visible gracias a los colores. Tras la asignación, se calcula la media de los puntos asignados al grupo y se actualiza el centroide con este valor. Se observa en la iteración dos como se han desplazados los centroides y se han reasignado los puntos al centro más cercano. Se sigue realizando estos pasos hasta que las asignaciones de los puntos no cambien. Esto representa el criterio de parada.

K-means es un algoritmo clásico de agrupamiento, siendo uno de los más populares debido a su simplicidad y eficiencia. En general K-means converge después de relativamente pocas iteraciones, pero a medida que el número de grupos, k, se hace más grande la eficiencia del algoritmo disminuye (Kanungo, 2002).

Uno de los principales inconvenientes que puede presentar K-means es la determinación del número de grupos, k, que se formarán a partir de los datos, dado que el algoritmo no entrega información sobre el mejor valor de k. Seleccionar un mal valor de k genera un resultado sub-óptimo y que puede no responder a las metas originales de la investigación. En general, el valor de k corresponde a un número entero relativamente pequeño. Existen varios métodos para tratar de encontrar un buen valor de k, por ejemplo, se realiza varias iteraciones del algoritmo con diferentes valores de k aleatorios y se elige el mejor k de acuerdo a alguna función de calidad. También, si se tiene algún conocimiento previo sobre la naturaleza de los documentos, se puede inferir algún número razonable de categorías a utilizar (Kanungo, 2002).

Al agrupar documentos con K-means, estos comienzan todos agrupados y luego son distribuidos en grupos más pequeños de documentos similares, realizándose varias iteraciones de este proceso hasta alcanzar el criterio de parada. El algoritmo K-means ha sido utilizado extensamente con documentos de textos, presentando una alta eficiencia.

Por las ventajas que posee este algoritmo, se decide escogerlo como base para el método a desarrollar para dar solución al problema planteado.

1.3.3.2 Método de Ward

El método de Ward es un procedimiento jerárquico en el cual, en cada etapa, se unen los dos grupos para los cuales se tenga el menor incremento en el valor total de la suma de los cuadrados de las diferencias, dentro de cada grupo, de cada individuo al centroide del grupo.

Siendo x^k_{ij} al valor de la j−ésima variable sobre el i−ésimo individuo del k−ésimo grupo, suponiendo que dicho grupo posee n_k individuos.

m^k al centroide del grupo k, con componentes m^k_j.

E_k a la suma de cuadrados de los errores del grupo k, o sea, la distancia euclídea al cuadrado entre cada individuo del grupo k a su centroide

$$\boldsymbol{E_k = \sum_{i=1}^{n_k}\sum_{j=1}^{n}(x_{ij}^k - m_j^k)^2 = \sum_{i=1}^{n_k}\sum_{j=1}^{n}(x_{ij}^k)^2 - n_k\sum_{j=1}^{n}(m_{ij}^k)^2}$$

E a la suma de cuadrados de los errores para todos los grupos, o sea, suponiendo que hay h grupos

$$\boldsymbol{E = \sum_{k=1}^{h} E_k}$$

El proceso comienza con m grupos, cada uno de los cuales está compuesto por un solo individuo, por lo que cada individuo coincide con el centro del grupo y por lo tanto en este primer paso se tendrá $E_k = 0$ para cada grupo y con ello, E = 0. El objetivo del método de Ward es encontrar en cada etapa aquellos dos grupos cuya unión proporcione el menor incremento en la suma total de errores, E (Gallardo San Salvador, 2015).

Suponiendo ahora que los grupos C_p y C_q se unen resultando un nuevo grupo C_t. Entonces el incremento de E es:

$$\Delta E_{pq} = E_t - E_p - E_q$$

$$= \left[\sum_{i=1}^{n_t}\sum_{j=1}^{n}\left(x_{ij}^t\right)^2 - n_t\sum_{j=1}^{n}\left(m_j^t\right)^2\right] - \left[\sum_{i=1}^{n_p}\sum_{j=1}^{n}\left(x_{ij}^p\right)^2 - n_p\sum_{j=1}^{n}\left(m_j^p\right)^2\right]$$

$$- \left[\sum_{i=1}^{n_q}\sum_{j=1}^{n}\left(x_{ij}^q\right)^2 - n_q\sum_{j=1}^{n}\left(m_j^q\right)^2\right] = n_p\sum_{j=1}^{n}\left(m_j^p\right)^2 + n_q\sum_{j=1}^{n}\left(m_j^q\right)^2 - n_t\sum_{j=1}^{n}(m_j^t)^2$$

Ahora bien

$$n_t m_j^t = n_p m_j^p + n_q m_j^q$$

de donde

$$n_t^2(m_j^t)^2 = n_p^2(m_j^p)^2 + n_q^2(m_j^q)^2 + 2n_p n_q m_j^p m_j^q$$

y como

$$2m_j^p m_j^q = (m_j^p)^2 + (m_j^q)^2 - (m_j^p - m_j^q)^2$$

se tiene

$$n_t^2(m_j^t)^2 = n_p(n_p + n_q)(m_j^p)^2 + n_q(n_p + n_q)(m_j^q)^2 - n_p n_q(m_j^p + m_j^q)^2$$

Dado que $n_t = n_p + n_q$, dividiendo por n^2_t se obtiene

$$(m_j^t)^2 = \frac{n_p}{n_t}(m_j^p)^2 + \frac{n_q}{n_t}(m_j^q)^2 - \frac{n_p n_q}{n_t^2}(m_j^p - m_j^q)^2$$

con lo cual se obtiene la siguiente expresión de ΔE_{pq}:

$$n_p\sum_{j=1}^{n}(m_j^p)^2 + n_q\sum_{j=1}^{n}(m_j^q)^2 - n_t\sum_{j=1}^{n}\left[\frac{n_p}{n_t}(m_j^p)^2 + \frac{n_q}{n_t}(m_j^q)^2 - \frac{n_p n_q}{n_t^2}(m_j^p - m_j^q)^2\right]$$

$$\Delta \mathrm{E}_{\mathrm{pq}} = n_p \sum_{j=1}^{n} \left(\boldsymbol{m}_j^p\right)^2 + n_q \sum_{j=1}^{n} \left(\boldsymbol{m}_j^q\right)^2 - m_p \sum_{j=1}^{n} \left(\boldsymbol{m}_j^p\right)^2 - n_q \sum_{j=1}^{n} \left(\boldsymbol{m}_j^q\right)^2 + \frac{n_p n_q}{n_t} \sum_{j=1}^{n} \left(\boldsymbol{m}_j^p - \boldsymbol{m}_j^q\right)^2$$

$$= \frac{n_p n_q}{n_t} \sum_{j=1}^{n} \left(\boldsymbol{m}_j^p - \boldsymbol{m}_j^q\right)^2$$

Así el menor incremento de los errores cuadráticos es proporcional a la distancia euclídea al cuadrado de los centroides de los grupos unidos. La suma E es no decreciente y el método, por lo tanto, no presenta los problemas de los métodos del centroide anteriores.

Para finalizar, cómo se pueden calcular los distintos incrementos a partir de otros calculados con anterioridad. Sea C_t el grupo resultado de unir C_p y C_q y sea C_r otro grupo distinto a los otros dos. El incremento potencial en E que se produciría con la unión de C_r y C_t es

$$\Delta \mathrm{E}_{\mathrm{rt}} = \frac{n_r n_t}{n_r + n_t} \sum_{j=1}^{n} \left(\boldsymbol{m}_j^r - \boldsymbol{m}_j^t\right)^2$$

Teniendo en cuenta que

$$\boldsymbol{m}_j^t = \frac{n_p m_j^p + n_q m_j^q}{n_t}$$

$$n_t = n_p + n_q$$

y la expresión

$$(m_j^t)^2 = \frac{n_p}{n_t}(m_j^p)^2 + \frac{n_q}{n_t}(m_j^q)^2 - \frac{n_p n_q}{n_t^2}(m_j^p - m_j^q)^2$$

se deduce

$$(m_j^r - m_j^t)^2 = (m_j^r)^2 + (m_j^t)^2 - 2 m_j^r m_j^t$$

$$= \left(m_j^r\right)^2 + \frac{n_p}{n_t}\left(m_j^p\right)^2 + \frac{n_q}{n_t}\left(m_j^q\right)^2 - \frac{n_p n_q}{n_t^2}\left(m_j^p - m_j^q\right)^2 - 2 m_j^r \frac{n_p m_j^p + n_q m_j^q}{n_t}$$

$$= \frac{n_p\left(m_j^r\right)^2 + n_q\left(m_j^r\right)^2}{n_t} + \frac{n_p}{n_t}\left(m_j^p\right)^2 + \frac{n_q}{n_t}\left(m_j^q\right)^2 - \frac{n_p n_q}{n_t^2}\left(m_j^p - m_j^q\right)^2$$

$$- 2 m_j^r \frac{n_p m_j^p + n_q m_j^q}{n_t} = \frac{n_p}{n_t}\left(m_j^r - m_j^p\right)^2 + \frac{n_q}{n_t}\left(m_j^r - m_j^q\right)^2 - \frac{n_p n_q}{n_t^2}\left(m_j^p - m_j^q\right)^2$$

con lo cual

$$\Delta E_{rt} = \frac{n_r n_t}{n_r + n_t} \sum_{j=1}^{n} \left(m_j^r - m_j^t\right)^2$$

$$= \frac{n_r n_t}{n_r + n_t} \sum_{j=1}^{n} \left[\frac{n_p}{n_t} \left(m_j^r - m_j^p\right)^2 + \frac{n_q}{n_t} \left(m_j^r - m_j^q\right)^2 - \frac{n_p n_q}{n_t^2} \left(m_j^p - m_j^q\right)^2 \right]$$

$$= \frac{n_r n_p}{n_r + n_t} \sum_{j=1}^{n} \left(m_j^r - m_j^p\right)^2 + \frac{n_r n_q}{n_r + n_t} \sum_{j=1}^{n} \left(m_j^r - m_j^q\right)^2$$

$$- \frac{n_r n_p n_q}{n_t (n_r + n_t)} \sum_{j=1}^{n} \left(m_j^p - m_j^q\right)^2$$

$$= \frac{1}{n_r + n_t} \sum_{j=1}^{n} \left[n_r n_p \left(m_j^r - m_j^p\right)^2 + n_r n_q \left(m_j^r - m_j^q\right)^2 - \frac{n_r n_p n_q}{n_p + n_q} \left(m_j^p - m_j^q\right)^2 \right]$$

A partir esto puede demostrar que la relación anterior se sigue verificando para una distancia que venga definida a partir de una norma que proceda de un producto escalar o que verifique la ley del paralelogramo (Murtagh, y otros, 2014).

1.3.3.3 Modelado de temas

En el aprendizaje automático y el procesamiento del lenguaje natural, un modelo temático es un tipo de modelo estadístico para descubrir los "temas" abstractos que ocurren en una colección de documentos. El modelado de temas es una herramienta de minería de texto de uso frecuente para el descubrimiento de estructuras semánticas ocultas en un cuerpo de texto. Intuitivamente, dado que un documento trata de un tema en particular, se podría esperar que algunas palabras aparezcan en el documento más o menos frecuentemente: "perro" y "hueso" aparecerán más a menudo en documentos sobre perros, "gato" y "miau" aparecerán en documentos sobre gatos, y "el" y "es" aparecerán igualmente en ambos. Un documento se refiere típicamente a múltiples temas en diferentes proporciones; por lo tanto, en un documento que es el 10% sobre los gatos y el 90% sobre los perros, probablemente habría unas 9 veces más palabras de perro que palabras de gato. Los "temas" producidos por las técnicas de modelado de tópicos son grupos de palabras similares. Un modelo de tema capta esta intuición en un marco matemático, que permite examinar un conjunto de documentos y descubrir, sobre la base de las estadísticas de las palabras en cada uno, cuáles son los temas y cuál es el balance de los temas de cada documento (Probabilistic Topic Models, 2012).

Los modelos temáticos también se denominan modelos temáticos probabilísticos, que se refieren a algoritmos estadísticos para descubrir las estructuras semánticas latentes de un cuerpo extenso de texto. En la era de la información, la cantidad de material escrito que se encuentra cada día está

simplemente más allá de la capacidad de procesamiento. Los modelos temáticos pueden ayudar a organizar y ofrecer conocimientos para comprender grandes colecciones de cuerpos de texto no estructurados. Originalmente desarrollado como una herramienta de minería de texto, se han utilizado modelos temáticos para detectar estructuras instructivas en datos tales como información genética, imágenes y redes. También tienen aplicaciones en otros campos como la bioinformática (Probabilistic Topic Models, 2012).

Modelo Latent Dirichlet Allocation (Asignación Latente de Dirichlet) (LDA)

LDA es un modelo generativo que permite que conjuntos de observaciones puedan ser modelados por grupos no observados que explican por qué algunos datos son similares.

Dado un documento, presuponemos que es una mezcla de categorías (o temas) y que las palabras del documento se deben a una categoría a la que debe pertenecer el documento. Se trata de una manera de descubrir automáticamente categorías o temas, a partir de una colección de documentos (Chen, 2016).

Este modelo representa los documentos como una mezcla de categorías, que genera palabras con cierta probabilidad. Asume que, a la hora de crear los documentos, estos se han generado de la siguiente manera:

- Se decide el número de palabras N que tiene el documento.
- Se escoge una mezcla de categorías a las que el documento pertenece, de acuerdo a una distribución Dirichlet de entre K categorías fijadas (por ejemplo 1/3 comida y 2/3 animales).

Se genera cada palabra de los documentos de la siguiente manera:

- Seleccionando una categoría de acuerdo a la distribución del anterior apartado.
- De acuerdo a la distribución multinomial de la categoría, se genera la palabra. Por ejemplo, si se escogió la categoría de comida, se genera la palabra “pizza” con un 25% de probabilidad, y con un 10% “sopa”, etc.

Asumiendo este modelo para una colección de documentos, LDA intenta volver hacia atrás para descubrir las categorías que generaron los documentos (Chen, 2016).

Aprendizaje

A partir de una colección de documentos D y fijado un número de categorías K que queremos descubrir. El algoritmo determina la representación de cada documento y las palabras asociadas a cada categoría:

- Asignar aleatoriamente a cada palabra una categoría.
- Por cada palabra W en el documento D, se asume que todas las asignaciones son correctas excepto esta. De esta forma, se actualiza la asignación actual de la palabra comprobando la asignación en otros documentos para la misma palabra, y la asignación de categorías en el documento.

Después de ejecutar lo anterior varias veces, se obtiene un estado en el que las asignaciones son bastante precisas.

1.3.4 Medidas de evaluación de agrupamiento

Una de las tareas más relevantes y complejas de la aplicación de algoritmos de agrupamiento es la evaluación de los resultados, conocida como validación de grupos. Dado que el agrupamiento es un proceso no supervisado, las medidas de desempeño usadas para el caso clasificación no se pueden utilizar (Huang, 2008).

Los resultados obtenidos por un algoritmo de agrupamiento dependen de las condiciones iniciales, tales como el número de grupos a crear para el caso de K-means. Para evaluar y seleccionar un esquema de agrupamiento, dos medidas resultan fundamentales: compactación y separación, es decir: se espera que los miembros dentro de cada grupo estén tan cerca como sea posible, y que los grupos entre sí estén ampliamente separados.

Un enfoque común para la evaluación de la calidad de los resultados de agrupamiento es usar índices de validación de grupos, los cuales tienen por objetivo encontrar un conjunto de grupos que se ajuste a una partición natural de los datos, usualmente definidos combinando compacidad y separabilidad (Huang, 2008).

Existen tres diferentes enfoques para estudiar la validez del resultado de un algoritmo de agrupamiento: criterios externos, criterios internos, criterios relativos.

Los criterios externos evalúan los resultados basados en una estructura de información pre-especificada, la cual es impuesta en el conjunto de datos, utilizando la intuición específica del usuario acerca de la estructura de grupos del conjunto de datos. El principal objetivo es determinar la calidad del algoritmo de agrupamiento para reconocer grupos existentes.

Los criterios internos estudian los resultados basados en las características propias de los grupos, sin información adicional acerca de los datos o repeticiones del proceso de agrupamiento. El objetivo primordial es determinarla calidad del algoritmo para generar particiones.

Los criterios relativos analizan los resultados basados en la comparación de esquemas de agrupamiento ejecutados en varias iteraciones, pero con diferentes valores de los parámetros de entrada o distintos subconjuntos de datos. Este enfoque no utiliza información adicional de los datos. El principal objetivo es determinar la calidad del algoritmo para generar grupos significativos.

En la mayoría de los casos prácticos no se tiene conocimiento previo sobre el conjunto de datos, por lo que aplicar criterios externos de validación de grupos resulta muy complejo. A partir de los experimentos realizados, los criterios internos resultaron ser más precisos que los criterios externos (Huang, 2008).

1.3.5 Visualización de los resultados

La visualización de resultados en la minería de texto, tiene crucial importancia para facilitar el descubrimiento de conocimiento, ya que entrega un panorama general de una gran cantidad de datos. Métodos no supervisados de minería de texto, como el agrupamiento, requieren un intenso trabajo que permita una correcta interpretación de los resultados con el fin de obtener información útil. Ejemplos de modelos para visualización de los resultados de un proceso de minería de texto son los dendrogramas, los gráficos de gotas y las nubes de términos

Un gráfico de gotas de colores es una representación visual de la distribución espacial esférica de la agrupación de comentarios resultantes de aplicar el algoritmo K-means. En este gráfico, se aprecian los comentarios identificados por un color, lo cual representa el grupo al que pertenecen.

Un dendrograma es un tipo de representación gráfica en forma de árbol que organiza los datos en subcategorías que se van dividiendo en otros hasta llegar al nivel de detalle deseado de las agrupaciones derivadas de la aplicación de un algoritmo de agrupamiento jerárquico.

Una nube de términos es un grupo de palabras clave etiquetadas en diferentes ubicaciones, formas, tamaños o colores, en forma de nube. Normalmente las de mayor tamaño y colores intensos, reflejan las temáticas de mayor importancia (por relevancia, volumen de contenido o actualidad) siendo las menos significativas aquellas más pequeñas y de colores más degradados. El principal objetivo de la nube de términos es facilitar al usuario la búsqueda de información relevante al indicarle las palabras más frecuentes en los grupos de comentario, representadas por esas etiquetas.

1.4 Tecnologías y herramientas

El lenguaje y las herramientas empleadas en el desarrollo del complemento, van en correspondencia con las utilizadas en el desarrollo de la aplicación AOpinion.

1.4.1 Lenguajes de programación

Python 2.7

Lenguaje interpretado o de script que tiene una sintaxis simple, clara y sencilla. Python usa tipado dinámico, pues no es necesario declarar el tipo de datos que va a contener una determinada variable.

Es multiplataforma y tiene gran cantidad de bibliotecas disponibles (González Duque, 2015). Se utilizan las siguientes bibliotecas:

- **BeautifulSoup:** es una biblioteca de Python para analizar documentos HTML. Esta biblioteca es útil para realizar web scraping (extracción de un sitio web) ya que crea un árbol con todos los elementos del documento y puede ser utilizado para extraer información de los sitios web (Richardson, 2012).

- **Request:** es una biblioteca para Python para el trabajo con HTTP, lo que permite obtener y trabajar con sitios web (Requests, 2014).

- **NLTK [5] (*Natural Language Toolkit*):** conjunto de técnicas que permiten el análisis y manipulación del lenguaje natural. Se utiliza para la creación de programas en Python que interpretan el lenguaje humano. Permite realizar tareas de transformación y limpieza de documentos tales como: eliminar caracteres especiales, signos de puntuación, convertir todo el texto en minúscula, eliminar palabras comunes o sin significado (conocidas como palabras de paradas) de la lengua en la que está escrito tales como: el, para, de, por, y, un, entre otras.

- **Sklearn:** es una biblioteca de código abierto para tareas de aprendizaje automático para el lenguaje de programación Python. Cuenta con varios algoritmos de clasificación, regresión y agrupación; entre los que se encuentran K-means y DBSCAN, y está diseñada para interoperar con las bibliotecas numéricas y científicas de Python, NumPy y SciPy.

- **Panda:** es una biblioteca de código abierto que proporciona estructuras de datos de alto rendimiento y fácil de usar y herramientas de análisis de datos para el lenguaje de

[5] Conjunto de técnicas disponible en http://www.nltk.org/

programación Python. En particular, ofrece estructuras de datos y operaciones para manipular tablas numéricas y series temporales.

- **Gensim:** es una biblioteca de código abierto de Python para modelado de temas, indexación de documentos y recuperación de similitudes. Utiliza NumPy, SciPy y opcionalmente Cython para el rendimiento. Está específicamente diseñado para manejar grandes colecciones de texto, utilizando streaming de datos y algoritmos incrementales eficaces, lo que lo diferencia de la mayoría de los paquetes de software científico que sólo se orientan al procesamiento por lotes y en memoria.

- **Numpy:** es el paquete fundamental para la computación científica con Python. Contiene entre otras cosas: un potente objeto de matriz N-dimensional, herramientas para integrar código C / C ++ y Fortran, álgebra lineal útil, transformada de Fourier y la capacidad de generar números aleatorios. También puede ser utilizado como un eficiente contenedor multidimensional de datos genéricos. Se pueden definir tipos de datos arbitrarios. Esto permite a NumPy integrarse de forma transparente y rápida con una amplia variedad de bases de datos.

- **SciPy:** es una biblioteca código abierto de herramientas y algoritmos matemáticos para Python. Contiene módulos para optimización, álgebra lineal, integración, interpolación, funciones especiales, procesamiento de señales y de imagen y otras tareas para la ciencia e ingeniería.

- **MatPlotLib:** es un módulo de dibujo de gráficas 2D para Python (matplotlib.org, 2015).

- **Psycopg[6]:** biblioteca que establece la conexión entre python y el gestor de base de datos PostgreSQL.

HTML 5.0

HTML 5 es una nueva versión con elementos, atributos, comportamientos y un conjunto amplio de tecnologías que permiten crear sitios web y diversas aplicaciones de gran alcance. Es un lenguaje de marca usado para estructurar y presentar el contenido para la web (Álvarez, Miguel Ángel, 2009).

[6] Biblioteca disponible en: https://pypi.python.org/pypi/psycopg2

En el presente trabajo se usa para la creación de las interfaces web que son mostradas al usuario.

CSS3

Las hojas de estilo en cascada (CSS por sus siglas en inglés) son las que ofrecen la posibilidad de definir las reglas y estilos de representación en diferentes dispositivos, ya sean pantallas de equipos de escritorio, portátiles, móviles, impresoras u otros dispositivos capaces de mostrar contenidos web. Permiten definir de manera eficiente la representación de las páginas, además es uno de los conocimientos fundamentales que todo diseñador web debe manejar a la perfección para realizar su trabajo (Frain, 2012).

JavaScript

JavaScript es un lenguaje que puede ser utilizado por profesionales y por quienes se inician en el desarrollo y diseño de sitios web. No requiere de compilación ya que el lenguaje funciona del lado del cliente, pues los navegadores son los encargados de interpretar estos códigos. Es muy utilizado para controlar la apariencia y manipular los eventos dentro de la ventana del navegador web, así como para validar datos de entrada en las interfaces de las aplicaciones (Pérez Valdés, Damián, 2007). La biblioteca utilizada de JavaScript en este trabajo es:

- **jQuery**[7] [8]: biblioteca de JavaScript en su versión 1.8.2 que permite programar páginas dinámicas compatibles con todos los navegadores. Facilita el desarrollo de aplicaciones enriquecidas del lado del cliente, en JavaScript, compatible con todos los navegadores.

1.4.2 Marco de trabajo

Un marco o ambiente de trabajo está orientado a la reutilización de componentes permitiendo el desarrollo rápido de aplicaciones. Los marcos de trabajo a emplear para la implementación del sistema son: Django 1.6 y Bootstrap 3.0.

Django 1.6.1

Django es un marco de trabajo web de código abierto escrito en Python que permite construir aplicaciones web más rápido y con menos código (Django, 2016).

---

[7] Manual de la biblioteca jQuery disponible en: http://www.desarrolloweb.com/manuales/manual-jquery.html

[8] Biblioteca disponible en: http://jquery.com/

Ventajas de Django:

- Aparte de las ventajas que tiene por ser marco de trabajo, Django promueve el desarrollo rápido, se construyen aplicaciones en cuestión de días y con el conocimiento suficiente esos días se pueden reducir a horas.
- Impulsa el desarrollo de código limpio al promover buenas prácticas de desarrollo web.
- Django usa una modificación de la arquitectura Modelo-Vista-Controlador (MVC), llamada MTV (Model - Template - View), que sería Modelo-Plantilla-Vista, esta forma de trabajar permite que sea pragmático.

Bootstrap 3.0

Bootstrap es una herramienta para crear interfaces de usuario totalmente adaptables a todo tipo de dispositivos y pantallas, sea cual sea su tamaño. Además, ofrece las herramientas necesarias para crear cualquier tipo de sitio web utilizando los estilos y elementos de sus bibliotecas. Es un soporte bastante bueno (casi completo) con HTML5 y CSS3, permitiendo ser usado de forma muy flexible para desarrollo web con unos excelentes resultados (Fontela, Alvaro).

1.4.3 Entorno de desarrollo integrado

Una de las herramientas que desempeña un papel importante en el desarrollo de soluciones informáticas son los Entornos de Desarrollo Integrado (IDE[9]). Estos ofrecen facilidades al equipo de desarrollo cuando se implementan las aplicaciones debido a que permite la identificación de errores comunes que se comenten a diario (Herramienta informática de Minería de Uso de la Web sobre los registros de navegación por Internet, 2010).

PyCharm 2016.3.2

Se decidió utilizar PyCharm 2016.3.2 para el desarrollo de la aplicación. Es un IDE o entorno de desarrollo integrado multiplataforma utilizado para desarrollar en el lenguaje de programación Python. Proporciona análisis de código y soporte para el desarrollo web con Django. Entre sus principales características se encuentra el autocompletado, el resaltador de sintaxis para código Python, HTML, CSS, JavaScript, así como para las plantillas de Django. PyCharm permite la integración de pruebas unitarias y múltiples opciones para refactorizar el código.

[9] Del inglés: Integrated Development Enviroment

PostgreSQL 9.4

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente (Martínez, 2013). Es usado para manejar grandes cantidades de información. Es multiplataforma, aporta flexibilidad y permite definir funciones personalizadas por medio de varios lenguajes (python, java, php) (Anton, César, 2015).

Se utiliza PgAdmin en su versión 1.20.0 como herramienta de código abierto con el propósito general de diseñar, mantener y administrar las bases de datos de PostgreSQL.

1.5 Conclusiones parciales

En este capítulo se presentaron determinadas definiciones para una mejor comprensión del tema. Se realizó el estudio de las tendencias actuales de las herramientas que emplean el análisis de comentarios de usuarios para determinar su funcionamiento. Por lo que al analizar los enfoques utilizados por los algoritmos de agrupamiento se identifican las principales características competitivas de la solución, en función del contexto para el cual es requerida.

El estudio de las tecnologías apropiadas al desarrollo de la solución, determinó emplear Python como lenguaje de programación en su versión 2.7 y, y PyCharm en su versión 2016.3.2 como entorno de desarrollo integrado para la implementación de la solución.

CAPÍTULO 2: Propuesta de solución

En el presente capítulo se describe brevemente cuáles son los pasos a seguir para desarrollar el proceso de solución propuesto, así como las métricas y algoritmos a que se utilizan. La metodología adoptada en este trabajo está basada en las etapas que caracterizan el modelo de proceso CRISP-DM: comprensión del ámbito de aplicación, comprensión de los datos, preparación de los datos, modelado, evaluación y desarrollo.

2.1 Metodología CRISP-DM

Una metodología consiste en un conjunto de actividades organizadas que tienen como objetivo la realización de un trabajo. Para cada actividad se define, además de las entradas y salidas, la forma en la que debe llevarse a cabo (Moine, 2013).

La metodología CRISP-DM es introducida en una amplia gama de profesiones y está descrita en términos de un modelo de proceso jerárquico, estructurada en seis fases (Olson, y otros, 2008), (Chapman, y otros, 2000). Algunas de estas fases son bidireccionales, lo que significa que permiten revisar parcial o totalmente las fases anteriores. El modelo consiste en:

Fase 1. Comprensión del ámbito de aplicación: esta fase se enfoca en la comprensión de los objetivos y requerimientos del proyecto desde una perspectiva no técnica, para luego convertir este conocimiento de los datos en la definición de un problema de minería de datos y en un plan preliminar diseñado para alcanzar los objetivos. Sus principales tareas son:

- Establecimiento de los objetivos del negocio: cuáles son, en el contexto inicial, los objetivos que se tienen y los criterios de éxito.
- Evaluación de la situación: se realiza un inventario de los factores que deban ser considerados para alcanzar el objetivo del análisis de datos y conformar el plan del proyecto.
- Establecimiento de los objetivos de la minería de datos: determina los resultados esperados en términos técnicos que posibilitan alcanzar los objetivos propuestos.
- Generación del plan del proyecto, herramientas, tecnologías y técnicas: describe paso a paso cómo se pretenden alcanzar los objetivos de la minería y por tanto del negocio.

Fase 2. Comprensión de los datos: esta fase comienza con la recopilación inicial de datos y continúa con las actividades que permiten familiarizarse con los datos y verificar su calidad. Las principales tareas de esta fase son:

- Recopilación inicial de datos: se realiza la adquisición de los datos de las fuentes identificadas. Se identifican los problemas encontrados y las soluciones que se le dan a estos problemas.
- Descripción de los datos: caracterización general de los datos, ya sea su formato, cantidad, llaves y cualquier otra información descubierta.
- Exploración de los datos: realizar un análisis simple de los datos, usando preguntas, visualización o técnicas de reporte. Se analizan descubrimientos preliminares, hipótesis y su impacto en el resto del proyecto.
- Verificación de calidad de datos: se analizan posibles errores que tengan los datos (si están completos, faltantes, diferencia de formato, etc.). Se determina cuáles son las soluciones a seguir para el tratamiento de los errores.

Fase 3. Preparación de datos: esta fase cubre todas las actividades necesarias para construir el conjunto de datos que es utilizado en las herramientas de modelado a partir de los datos en bruto iniciales. Las tareas incluyen:

- Selección de atributos: se decide qué parte de los datos son incluidos o excluidos en el análisis y el porqué de esta selección (relevancia, calidad, restricciones, etc.).
- Limpieza de datos: su objetivo es elevar la calidad de los datos al nivel requerido por las técnicas de análisis seleccionadas (estimación de datos faltantes, decisiones tomadas para eliminar los problemas de la etapa anterior, transformaciones realizadas, etc.).
- Construcción de datos: se pueden crear atributos derivados a partir de atributos existentes o también generar registros, que no existen en la base datos almacenados, pero son válidos incluir en la modelación.
- Integración de datos: se mezclan datos de registros de diferentes tablas con información sobre el mismo objeto. Se pueden incluir datos agregados o dependientes de varios registros.
- Formateo de datos: se modifican los datos de forma tal que no cambie su significado pero que facilite la modelación.

Fase 4. Modelado: en esta fase varias técnicas de modelado son seleccionadas y aplicadas. También se realiza el diseño para la posterior evaluación del modelo. Las tareas principales con que cuenta esta fase son:

- Selección de la técnica de modelado: se escoge cuál técnica de modelación específica se va a utilizar (árboles de decisión, agrupamiento, redes neuronales, etc.). Se documenta la técnica escogida y el conjunto de requerimientos que deben cumplir los datos.
- Diseño de la evaluación: permite probar la validez y calidad del modelo. Se debe decidir cómo dividir el conjunto de datos, en conjunto de entrenamiento, prueba y validación.
- Construcción del modelo: se ejecuta la herramienta de modelación, se obtiene el modelo real que produce la herramienta de modelación y se realiza una interpretación del modelo resultante, describiendo las dificultades encontradas en la interpretación.
- Evaluación del modelo: se evalúa el modelo obtenido según el conocimiento del negocio y los criterios de éxito de minería de datos planteados, se realiza una comparación de la calidad de los modelos construidos. Según el resultado de la evaluación, se ajustan los parámetros y se vuelve a la etapa o fase anterior, hasta obtener los resultados esperados.

Fase 5. Evaluación: se evalúan los resultados finales obtenidos del modelo y se revisan los pasos del proceso realizado. Luego se compara el modelo obtenido con los objetivos inicialmente planteados. Las tareas de esta fase son:

- Evaluación de resultados: se evalúan los resultados del modelo desde el punto de vista de la exactitud técnica y del negocio. Los modelos aprobados son los que permiten alcanzar con éxito los objetivos propuestos.
- Revisar el proceso: se realiza la revisión del proceso resaltando si algo debe repetirse.
- Establecimiento de los siguientes pasos o acciones: de acuerdo con las pruebas realizadas hay que decidir cómo continuar, si finalizar o realizar nuevas iteraciones. Depende del cumplimiento de los objetivos y de los recursos disponibles.

Fase 6. Despliegue: en esta fase el conocimiento obtenido es organizado y presentado de modo que el que el usuario final pueda usarlo. Se establece una planificación de la monitorización y del mantenimiento del proceso, se generan los informes finales y se realiza la revisión del proyecto. Las tareas de esta fase son:

- Planificación de despliegue: según los resultados obtenidos en la evaluación, se decide una estrategia a seguir, sus pasos y como realizarlos.
- Planificación de la monitorización y del mantenimiento: una correcta estrategia de mantenimiento evita utilizar resultados incorrectos por largos períodos de tiempo.

- Generación de informe final: se debe organizar y resumir los resultados y experiencias del desarrollo del proyecto, los cuales son presentados en forma de reporte técnico, memoria, artículo, etc.
- Revisión del proyecto: definir qué estuvo bien, qué estuvo mal y qué se puede hacer mejor, es decir, resumir experiencias importantes en el desarrollo del proyecto.

En la figura 5 se muestra el proceso CRISP-DM, donde la secuencia de las fases no es rígida, ya que el movimiento entre fases diferentes es siempre requerido. Las flechas indican las dependencias más importantes y frecuentes entre fases.

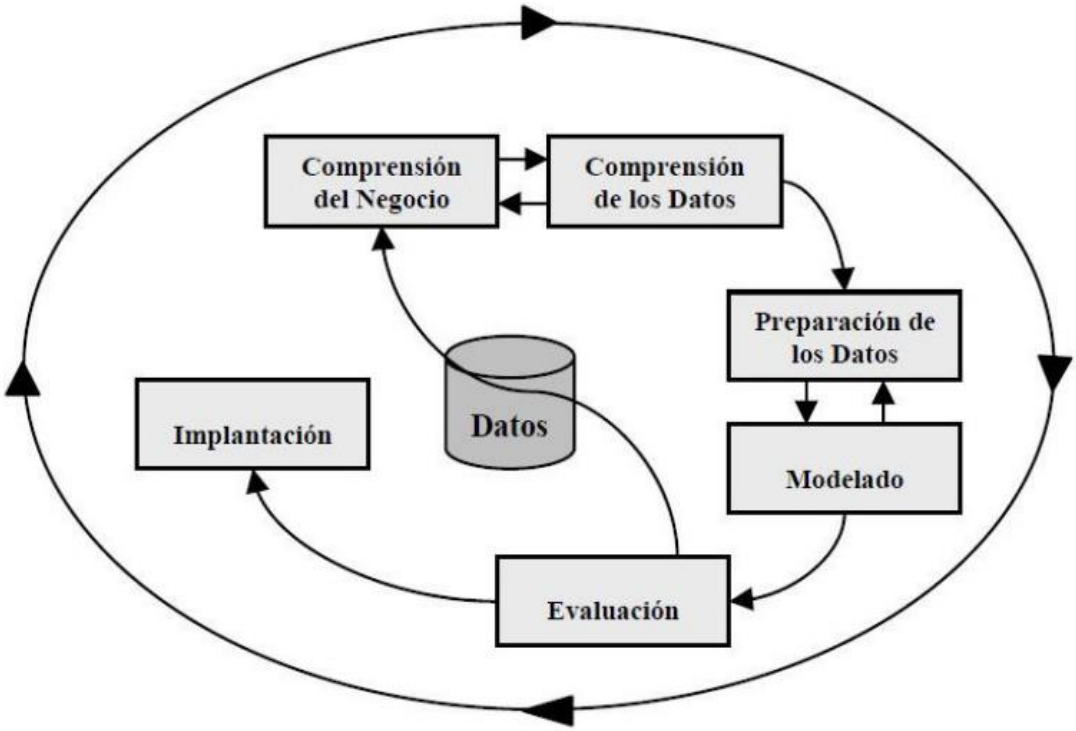

Figura 5 Etapas del proceso CRISP-DM (Chapman, y otros, 2000)

CRISP-DM puede ser visto como una implementación del proceso KDD, guiando a los usuarios en la puesta en práctica de la minería en sistemas reales. En la figura siguiente (Figura 6) se puede observar las correspondencias entre las etapas de KDD y CRISP-DM, donde la etapa de comprensión del ámbito de aplicación puede ser identificada con el desarrollo del entendimiento del dominio de aplicación, el conocimiento previo relevante y las metas de los usuarios finales, mientras que la etapa de desarrollo puede ser identificada con la consolidación a través de la incorporación del conocimiento al sistema (Azevedo, y otros, 2008).

KDD	CRISP-DM
Pre KDD	Comprensión del Ámbito de Aplicación
Selección	Comprensión de los Datos
Pre Procesamiento	
Transformación	Preparación de los Datos
Data Mining	Modelado
Interpretación/Evaluación	Evaluación
Post KDD	Desarrollo

Figura 6 Resumen correspondencias entre KDD y CRISP-DM (Azevedo, y otros, 2008)

Para el desarrollo de este trabajo se hizo necesario adaptar las fases y tareas de la metodología CRISP-DM al problema de esta investigación. Como se menciona anteriormente, se desarrolla un proceso de minería de texto, el cual tiene un flujo de trabajo aceptado por la comunidad internacional.

En el próximo epígrafe se explican las distintas fases de la metodología CRISP-DM aplicadas a esta investigación.

2.2 Comprensión del ámbito de aplicación del estudio

El contexto del estudio es el análisis de comentarios de usuarios, los cuales constituyen una ayuda para que los integrantes de los equipos de desarrollo conozcan las opiniones de los clientes, relacionados con las funcionalidades del software, permitiéndoles mejorar el producto y erradicar no conformidades. Estos comentarios, son publicados en los blogs de la Universidad de las Ciencias Informáticas.

Se ha decidido desarrollar un método que permita la obtención de información relevante a partir de estos comentarios, para apoyar al proceso de desarrollo de software en la UCI.

Se revisa la factibilidad de obtener los recursos necesarios para llevar a cabo este trabajo, como lo son la disponibilidad de un volumen considerable de comentarios de usuarios. Por otro lado, se entiende que el estudio presenta ciertas restricciones, dado que no es abundante la cantidad de comentarios publicados en los blogs de la universidad.

2.3 Comprensión de los datos

En esta etapa se seleccionan los blogs que son parte del estudio, de los cuales se extraen los datos (en esta investigación son comentarios de usuarios). Se escogen los que contienen publicaciones relacionadas con el desarrollo de software, entre ellos, los más significativos son:

- www.dragones.uci.cu

- www.android.uci.cu
- www.humanos.uci.cu
- www.php.uci.cu
- www.firefoxmania.uci.cu

Dichas fuentes contienen artículos, de los cuales se extraen los comentarios. Cada comentario contiene un autor, una fecha y un contenido. Seguidamente se transforman los datos a un formato común, unificando toda la información recogida en un archivo .txt con dicho formato (cada línea del archivo se considera un comentario independiente).

Para esta investigación, solo se tiene en cuenta el contenido del mismo.

2.3.1 Pseudocódigo del algoritmo ScrapWeb

Entrada del algoritmo: URL

Salida: Fichero .txt

Aclaraciones: La URL de entrada debe ser de alguno de los sitios soportados y de páginas web que contengan comentarios de usuarios.

Crear una variable *H* para guardar el código html y otra *C* para guardar los comentarios.

Si sitio es soportado **Entonces**

Capturar html del sitio en *H*

Para cada comentario en *H* **Hacer**

Guardar comentario en *C*

Fin Para

Guardar comentarios de *C* en fichero *.txt*

FinSi

Si no Hacer

Mostrar mensaje de error

FinSi no

2.3.2 Pseudocódigo del algoritmo Cargar Datos

Entrada: Fichero .txt

Salida: Diccionario de comentarios

Crear un diccionario *D* para almacenar los comentarios

Para cada línea del fichero *.txt* **Hacer**

Transformar a codificación utf-8

Almacenar la línea en la variable *D*

Fin Para

Retornar *D*

2.4 Preparación de los datos

En esta fase se procesan los comentarios aplicándole las siguientes técnicas de procesamiento del lenguaje natural (Ver epígrafe 1.3):

- Se eliminan todos los caracteres no alfanuméricos del texto como los signos de puntuación.
- Se convierten todas las palabras en minúsculas.
- Se eliminan las palabras de paradas, (preposiciones, conjunciones, artículos, etc).
- Se aplica stemming (radicalizar) con el fin de reducirlas a su raíz.

Se crea la matriz documento-término que es una representación de las similitudes y diferencias entre de los comentarios en un espacio vectorial (Ver epígrafe 1.3) y que sirve como entrada para los algoritmos de agrupamientos aplicados.

2.4.1 Pseudocódigo del algoritmo de pre-procesamiento

Entrada: Diccionario con los comentarios en *D*
Salida: Comentarios pre-procesados en *P*

Crear diccionario *P* para guardar los comentarios procesados, una *T* para guardar lista de tokens
Para cada comentario en *D* **Hacer**

Transformar texto a minúscula
Eliminar signo de puntuación
Eliminar palabras de parada
Transformar en lista de tokens y **guardar** en *T*
Radicalizar *T*

Guardar *T* en *P*

Fin Para

Retornar *P*

2.4.2 Pseudocódigo del algoritmo para la creación de la Matriz TD

Entrada: Diccionario de comentarios *P*

Salida: Matriz *TD*

Crear vectorizador *V*

//Es necesario para transformar los comentarios a vectores

Para cada comentario en *P* **Hacer**

Transformar a vector usando *V*

Almacenar vector en *TD*

Fin Para

Retornar *TD*

2.5 Modelado

En una primera etapa se requiere formar un conjunto de datos con los comentarios pre-procesados, y transformados en forma de matriz documento-término, por lo que se debe encontrar un modelo de agrupamiento que permita distinguir los grupos presente en los comentarios.

2.5.1 Aplicación de métodos de aprendizaje no supervisados

Se aplica la técnica de agrupamiento con una variante particional (K-means), una variante jerárquica (Ward-cluster) y el algoritmo LDA para realizar el modelado de temas. Para el correcto trabajo con los primeros dos algoritmos se utilizan dos medidas de similitud (ver epígrafe 1.3.3).

2.5.2 Pseudocódigo del algoritmo K-means

Entrada: Matriz *TD*

Salida: Comentarios agrupados en *C*

Definir cantidad de grupos

Repetir

Calcular los centroides de los grupos *Zj*

Redistribuir patrones entre los grupos

// Esto se realiza utilizando la mínima distancia euclídea al cuadrado como clasificador

Hasta que no cambie el valor de los centroides

Retornar *C*

2.5.3 Pseudocódigo del algoritmo Ward-Cluster

Entrada: Matriz *TD*

Salida: Dendrograma *A*

Seleccionar etiquetas

Detectar valores atípicos

Elegir medida de similitud

Obtener matriz distancia

Repetir

    Buscar grupos más similares

    Unir estos grupos en uno nuevo

    Calcular distancia entre este grupo y el resto

Hasta que todos los objetos estén en un mismo grupo

Retornar *A*

2.5.4 Pseudocódigo del algoritmo LDA

Entrada: Diccionario de comentarios *D*

Salida: Listado de términos ordenados por frecuencia *L*

Pre-procesar *D*

Crear el corpus *T* a partir e *D*

Obtener listado de términos ordenados por frecuencia *L*

Retornar *L*

2.6 Evaluación

Evaluar el rendimiento de un algoritmo de agrupamiento no es tan trivial como contar el número de errores o la precisión y el recuerdo de un algoritmo de clasificación supervisado. En particular, cualquier métrica de evaluación no debe tener en cuenta los valores absolutos de los rótulos de

agrupamiento, sino más bien si este agrupamiento define separaciones de los datos similares a algún conjunto de verdades fundamentales o satisfacer alguna suposición tal que los miembros pertenecen a la misma clase son más similares que los miembros de diferentes clases de acuerdo a alguna medida de similitud (Scikit-learn, 2017).

2.6.1 Índice Rand ajustado (ARI)

Dado el conocimiento de las asignaciones de etiquetas verdaderas (labels_true) y nuestras asignaciones de algoritmos de agrupación de las mismas muestras (labels_pred), el índice de Rand ajustado es una función que mide la similitud de las dos asignaciones, ignorando las permutaciones y con normalización casual (Scikit-learn, 2017).

```
>>> from sklearn import metrics
>>> labels_true = [0, 0, 0, 1, 1, 1]
>>> labels_pred = [0, 0, 1, 1, 2, 2]

>>> metrics.adjusted_rand_score(labels_true, labels_pred)
0.24...
```

Uno puede permutar 0 y 1 en las etiquetas previstas, renombrar 2 a 3, y obtener la misma puntuación:

```
>>> labels_pred = [1, 1, 0, 0, 3, 3]
>>> metrics.adjusted_rand_score(labels_true, labels_pred)
0.24...
```

Por otra parte, la puntuación ajustada es simétrica: cambiar el argumento no cambia la puntuación. Por lo tanto, puede utilizarse como una medida de consenso:

```
>>> metrics.adjusted_rand_score(labels_pred, labels_true)
0.24...
```

El etiquetado perfecto se puntúa 1.0:

```
>>> labels_pred = labels_true[:]
>>> metrics.adjusted_rand_score(labels_true, labels_pred)
1.0
```

Las malas puntuaciones (por ejemplo, etiquetados independientes) tienen valores negativos o cercanos a 0.0:

```
>>> labels_true = [0, 1, 2, 0, 3, 4, 5, 1]
>>> labels_pred = [1, 1, 0, 0, 2, 2, 2, 2]
>>> metrics.adjusted_rand_score(labels_true, labels_pred)
-0.12...
```

Ventajas

- Las asignaciones de etiquetas aleatorias (uniformes) tienen una puntuación ARI cercana a 0.0 para cualquier valor de n_clusters y n_samples.

- Rango limitado [-1, 1]: los valores negativos son malos (etiquetados independientes), los agrupamientos similares tienen una ARI positiva, 1,0 es la puntuación perfecta.

- No se hace ninguna suposición sobre la estructura del grupo: puede utilizarse para comparar algoritmos de agrupamiento como K-means que asume formas isotrópicas con resultados de algoritmos de agrupación espectral que pueden encontrar grupos con formas "dobladas".

Inconvenientes

- ARI requiere conocimiento de las etiquetas verdaderas, mientras que casi nunca está disponible en la práctica o requiere asignación manual por anotadores humanos (como en el aprendizaje supervisado).

Sin embargo, ARI también puede ser útil en un entorno puramente sin supervisión como un componente para un índice de consenso que se puede utilizar para la selección de modelo de agrupación.

Formulación matemática

Si C es una asignación de etiqueta verdadera y K la agrupación, se define a y b como:

a, el número de pares de elementos que están en el mismo conjunto en C y en el mismo conjunto en K.

b, el número de pares de elementos que están en conjuntos diferentes en C y en conjuntos diferentes en K.

El índice de Rand bruto (no ajustado) es entonces dado por:

$$RI = \frac{a + b}{C_2^{n_{samples}}}$$

Donde $C_2^{n_{samples}}$ es el número total de pares posibles en el conjunto de datos (sin ordenar).

Sin embargo, la puntuación RI no garantiza que las asignaciones de etiquetas aleatorias obtengan un valor cercano a cero (especialmente si el número de grupos está en el mismo orden de magnitud que el número de muestras).

Para contrarrestar este efecto podemos descartar el esperado $RI\ E[\boldsymbol{RI}]$ de etiquetados aleatorios definiendo el índice Rand ajustado como sigue:

$$ARI = \frac{RI - E[RI]}{\max(RI) - E[RI]}$$

2.6.2 Homogeneidad, Integridad y V-measure

Dado el conocimiento de las asignaciones de etiquetas verdaderas de las muestras, es posible definir alguna métrica intuitiva usando análisis de entropía condicional (Scikit-learn, 2017).

En particular, Rosenberg y Hirschberg (2007) definen los siguientes dos objetivos deseables para cualquier asignación de grupo:

- Homogeneidad: cada grupo contiene sólo miembros de una sola clase.
- Completitud: todos los miembros de una clase dada se asignan al mismo grupo.

Podemos convertir esos conceptos como puntuaciones homogeneity_score y completeness_score. Ambos están limitados por 0,0 y por encima de 1,0 (mayor es mejor):

```
>>> from sklearn import metrics
>>> labels_true = [0, 0, 0, 1, 1, 1]
>>> labels_pred = [0, 0, 1, 1, 2, 2]

>>> metrics.homogeneity_score(labels_true, labels_pred)
0.66...

>>> metrics.completeness_score(labels_true, labels_pred)
0.42...
```

Su media armónica llamada V-measure es calculada por v_measure_score:

```
>>> metrics.v_measure_score(labels_true, labels_pred)
0.51...
```

La Homogeneidad, la Integridad y la V-measure se pueden calcular a la vez usando homogeneity_completeness_v_measure como sigue:

```
>>> metrics.homogeneity_completeness_v_measure(labels_true, labels_pred)
...
(0.66..., 0.42..., 0.51...)
```

La siguiente asignación de agrupamiento es ligeramente mejor, ya que es homogénea pero no completa:

```
>>> labels_pred = [0, 0, 0, 1, 2, 2]
>>> metrics.homogeneity_completeness_v_measure(labels_true, labels_pred)
...
(1.0, 0.68..., 0.81...)
```

V_measure_score es simétrico: puede utilizarse para evaluar el acuerdo de dos asignaciones independientes en el mismo conjunto de datos.

Este no es el caso de completeness_score y homogeneity_score: ambos están vinculados por la relación:

```
homogeneity_score(a, b) == completeness_score(b, a)
```

Ventajas

- Puntuaciones Limitadas: 0.0 es tan malo como puede ser, 1.0 es una puntuación perfecta.
- Interpretación intuitiva: agrupación con mala V-measure puede ser cualitativamente analizado en términos de homogeneidad y completitud para sentir mejor qué tipo de errores se realiza por la asignación.

No se hace ninguna suposición sobre la estructura del grupo: puede utilizarse para comparar algoritmos de agrupamiento como K-means que asume formas de blob isotrópicas con resultados de algoritmos de agrupación espectral que pueden encontrar grupos con formas "dobladas".

Inconvenientes

- Las métricas introducidas previamente no se normalizan en lo que respecta al etiquetado aleatorio: esto significa que, dependiendo del número de muestras, grupos y etiquetas verdaderas, un etiquetado completamente aleatorio no siempre proporciona los mismos valores de homogeneidad, completitud y, por tanto, V-measure. En particular, el etiquetado aleatorio no da resultado cero, especialmente cuando el número de grupos es grande. Este problema se puede ignorar con seguridad cuando el número de muestras es más de mil y el número de grupos es menor que 10. Para tamaños de muestra más pequeños o mayor número de grupos es más seguro utilizar un índice ajustado como el índice de Rand ajustado (ARI).

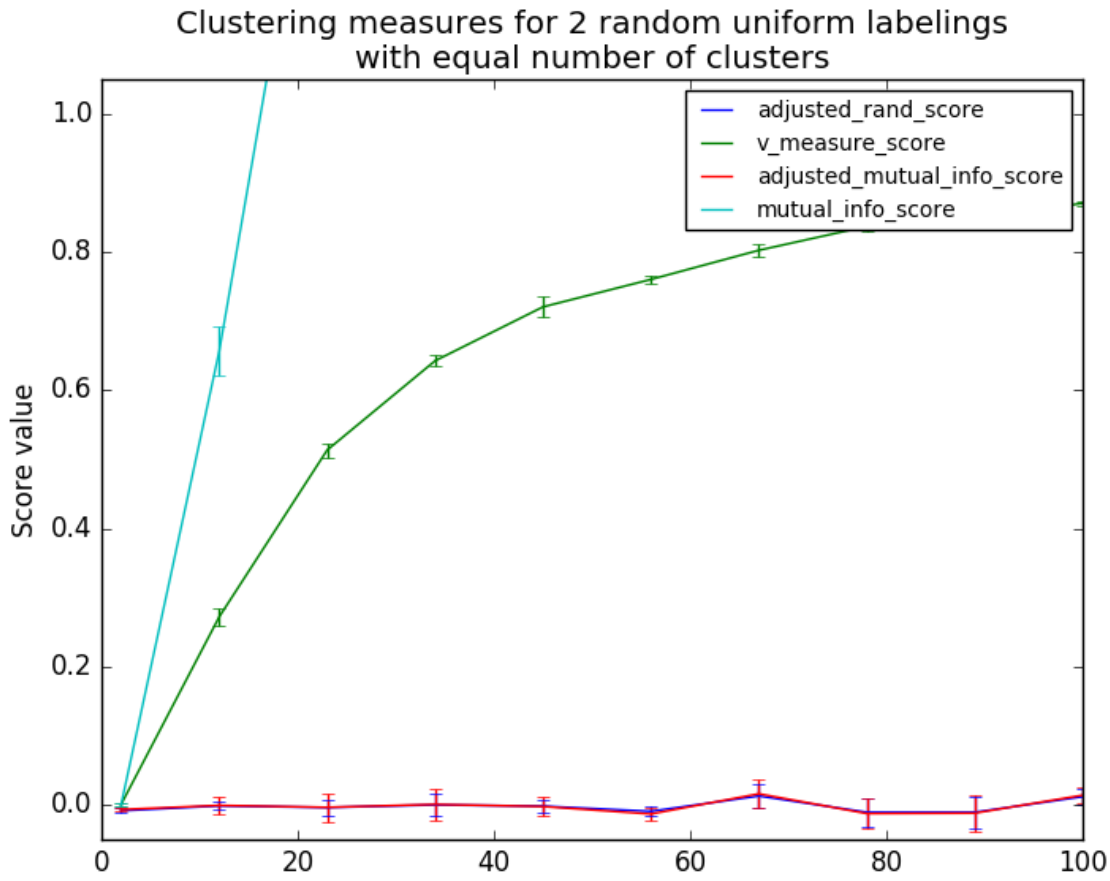

Figura 7:Comparación entre diferentes métricas (Scikit-learn, 2017)

Estas métricas requieren el conocimiento de las etiquetas verdaderas mientras que casi nunca están disponibles en la práctica o requieren asignación manual por anotadores humanos (como en el entorno de aprendizaje supervisado).

Formulación matemática

Las puntuaciones de homogeneidad y completitud son formalmente dadas por:

$$h = 1 - \frac{H(C|K)}{H(C)}$$

$$c = 1 - \frac{H(K|C)}{H(K)}$$

Donde H (C | K) es la entropía condicional de las clases dadas las asignaciones del grupo y está dada por:

$$H(C|K) = -\sum_{c=1}^{|C|}\sum_{k=1}^{|K|}\frac{n_{c,k}}{n}\cdot\log\left(\frac{n_{c,k}}{n_k}\right)$$

y H (C) es la entropía de las clases y está dada por:

$$H(C) = -\sum_{c=1}^{|C|} \frac{n_c}{n} \cdot \log\left(\frac{n_c}{n}\right)$$

siendo **n** el número total de muestras, n_c y n_k el número de muestras pertenecientes respectivamente a la clase c y el grupo k, y finalmente $n_{c,k}$ el número de muestras de la clase c asignadas al grupo k (Scikit-learn, 2017).

La entropía condicional de los grupos de clase H (K | C) y la entropía de los grupos H (K) se definen de forma simétrica.

Rosenberg y Hirschberg definen más la V-measure como la media armónica de homogeneidad y completitud:

$$v = 2 \cdot \frac{h \cdot c}{h + c}$$

2.6.3 Coeficiente de silueta

Si no se conocen las etiquetas verdaderas, la evaluación debe realizarse utilizando el propio modelo. El Coeficiente de Silueta (sklearn.metrics.silhouette_score) es un ejemplo de tal evaluación, donde una puntuación de Coeficiente de Silueta mayor se relaciona con un modelo con grupos mejor definidos. (Scikit-learn, 2017) El Coeficiente Silueta se define para cada muestra y se compone de dos puntuaciones:

a: La distancia media entre una muestra y todos los demás puntos de la misma clase.

b: Distancia media entre una muestra y todos los demás puntos en el siguiente grupo más cercano.

El Coeficiente de Silueta **s** para una muestra simple se da entonces como:

$$s = \frac{b - a}{\max(a, b)}$$

El Coeficiente de Silueta para un conjunto de muestras se da como la media del Coeficiente de Silueta para cada muestra.

```
>>> from sklearn import metrics
>>> from sklearn.metrics import pairwise_distances
>>> from sklearn import datasets
>>> dataset = datasets.load_iris()
>>> X = dataset.data
>>> y = dataset.target
```

En el uso normal, el coeficiente de silueta se aplica a los resultados de un análisis de conglomerados.

```
>>> import numpy as np
>>> from sklearn.cluster import KMeans
>>> kmeans_model = KMeans(n_clusters=3, random_state=1).fit(X)
>>> labels = kmeans_model.labels_
>>> metrics.silhouette_score(X, labels, metric='euclidean')
...
0.55...
```

Ventajas

- La puntuación está limitada entre -1 para la agrupación incorrecta y +1 para la agrupación altamente densa. Las puntuaciones en torno a cero indican que los grupos se superponen.
- La puntuación es mayor cuando los grupos son densos y bien separados, lo que se relaciona con un concepto estándar de un grupo.

Inconvenientes

El Coeficiente de Silueta es generalmente más alto para los grupos convexos que otros conceptos de grupos, tales como grupos basados en densidad como los obtenidos a través de DBSCAN.

2.7 Desarrollo

A continuación, se expone en detalle cada uno de los componentes de este método, así como las relaciones existentes entre los mismos.

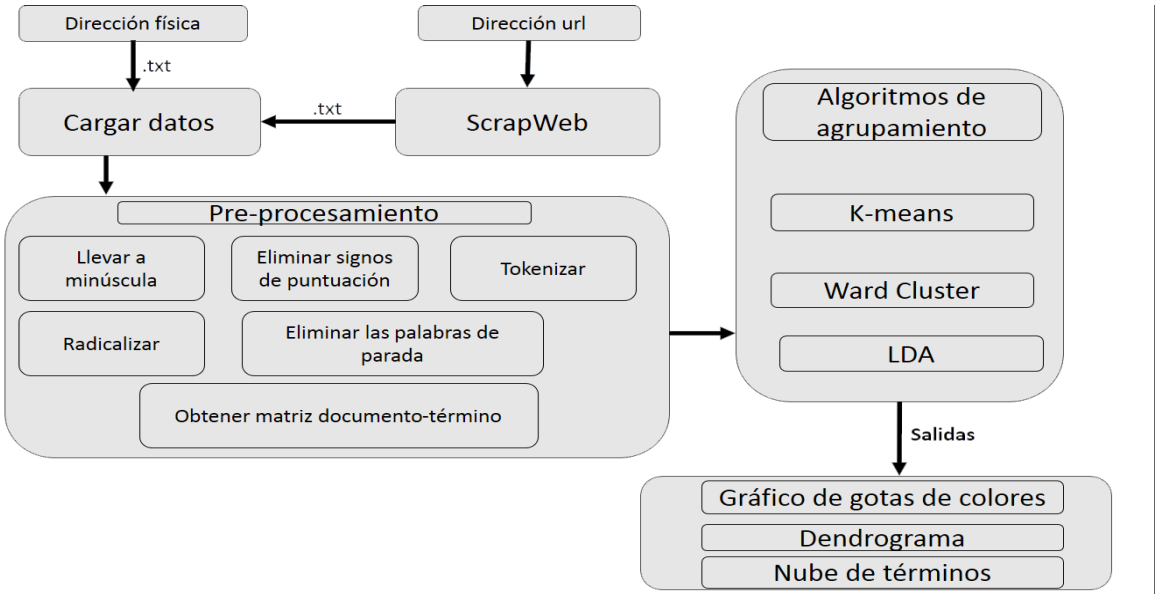

Figura 8 Esquema del método propuesto. (Elaboración propia)

El método propuesto tiene dos vías para recibir entradas: la primera es a través de una dirección url de un sitio (soportado) que contenga comentarios de usuarios para extraerlos y luego almacenarlo en un fichero .txt; la segunda es la dirección de un fichero .txt almacenado; donde en dicho fichero, cada línea representa un comentario. Estos ficheros .txt representan el conjunto de datos (comentarios de usuarios).

Luego se procede a realizar el pre-procesamiento de los comentarios de usuarios presentes en el conjunto de datos; tarea compuesta por los métodos llevar_Minúscula, eliminar_Puntuación, obtener_Tokens, obtener_Stem y obtener_MatrizTD.

Posteriormente se realiza el agrupamiento mediante los algoritmos K-means, Ward-cluster y LDA.

Finalmente se obtiene como resultado los grupos generados por K-means con un gráfico que muestra la composición de los grupos, una nube de términos generada por LDA y un dendrograma como resultado de Ward-cluster.

2.7.1 Salidas del método implementado

Al aplicar el método implementado se obtiene tres formas de visualización, lo cual ayuda en la interpretación, contrastación y comparación de los resultados lo que permite un conocimiento en profundidad y detalle de los mismos, de tal forma que se transformen en información comprensible para el usuario.

Gráfico de gotas de colores

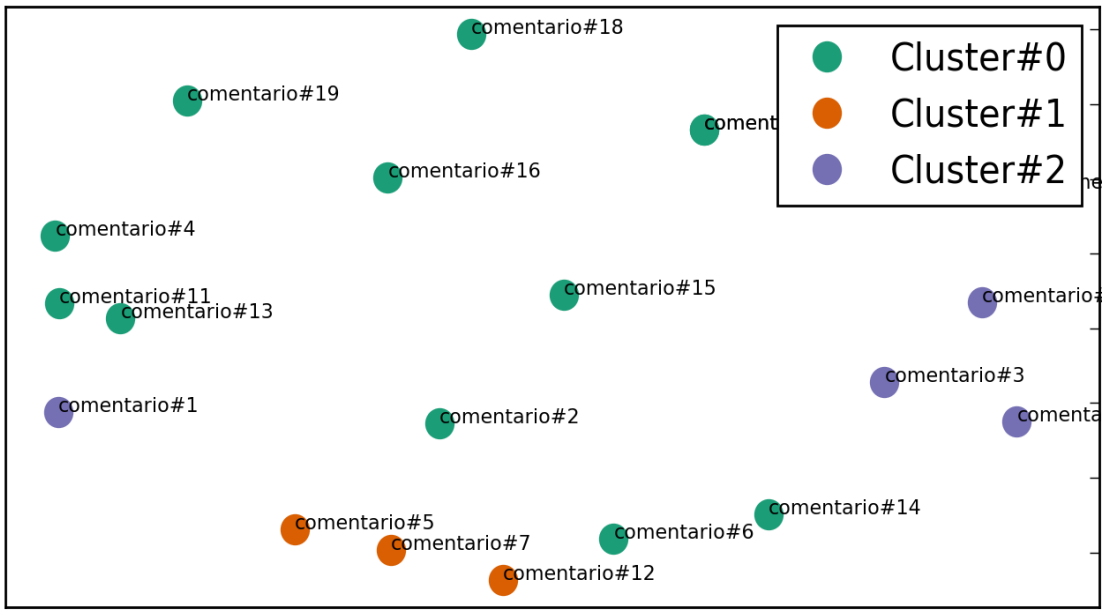

Figura 9 Ejemplo de gráfico de gotas resultante de K-means. (Elaboración propia)

Dendrograma

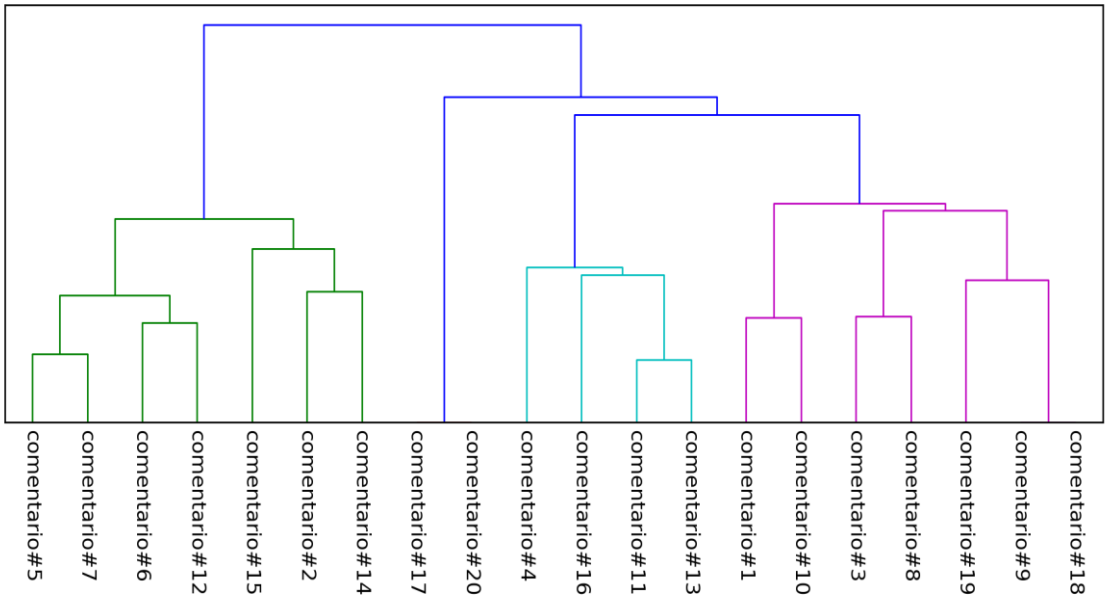

Figura 10 Ejemplo de dendrograma. (Elaboración propia)

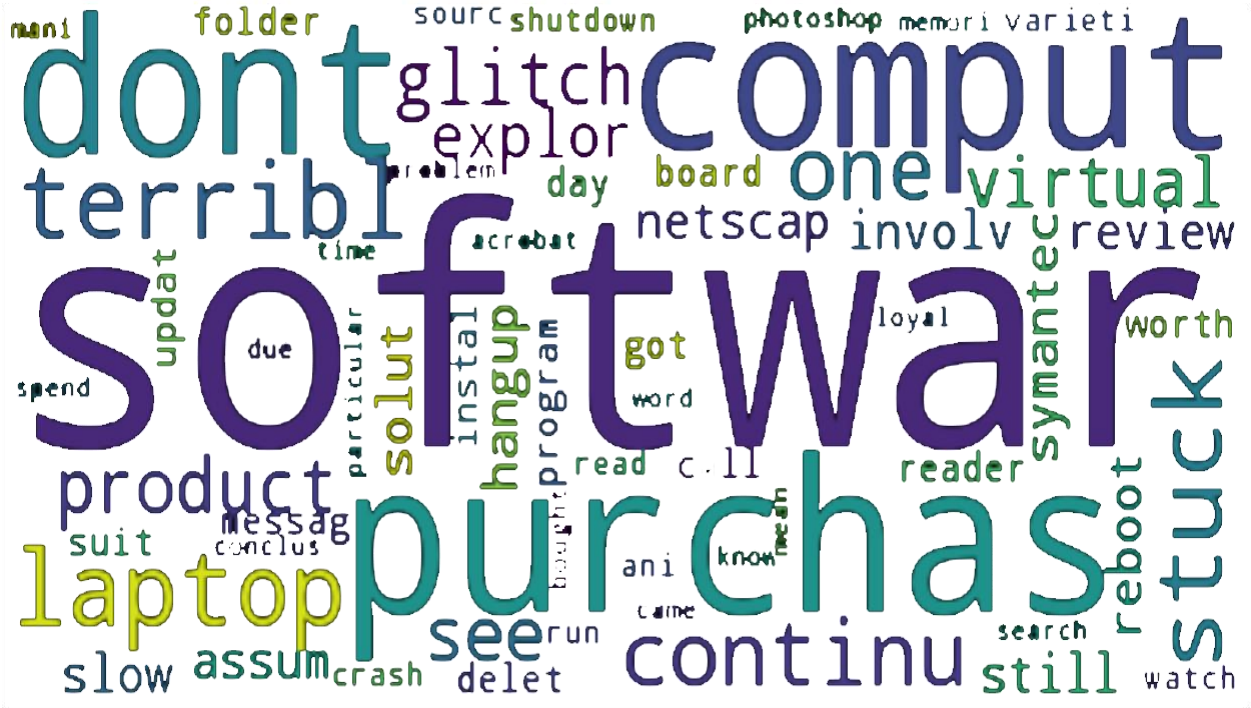

Figura 11 Ejemplo de nube de términos. (Elaboración propia)

2.8 Conclusiones del capítulo

En este capítulo se ha descrito la solución para la detección de información relevante de los comentarios de usuarios de aplicaciones de software en la UCI, siguiendo la propuesta elaborada previamente en la investigación. Se han enunciado diversas definiciones que permitieron formalizar las variables del estudio, lo cual condujo el proceso de desarrollo de los algoritmos de pre-procesamiento y de agrupamiento, concluyendo además que:

- La obtención de la solución para la detección de información relevante de los usuarios sobre las aplicaciones de software en la UCI, posibilitó incorporar un nuevo instrumento para el análisis de comentarios de usuarios.
- La implementación de la solución diseñada a partir del algoritmo para el agrupamiento de texto por temáticas según los comentarios de usuarios en el proceso de desarrollo de aplicaciones de software, facilitó el estudio y la comprensión del comportamiento de los comentarios de usuarios acerca de las aplicaciones de software.
- La solución fue implementada en Python para el sistema AOpinion, lo cual permitió acceder a los beneficios que brinda esta plataforma.

CAPÍTULO 3. Validación de la solución

3.1 Introducción

En el presente capítulo se realiza la validación de la solución propuesta. La precisión del algoritmo de agrupamiento se mide a partir de métricas de evaluación de rendimiento que comprueban la eficiencia con que se ejecutan, y la calidad de las soluciones que éstos producen. Con el fin de comprobar la utilidad del método, se utilizó el Blog HumanOS[10] de la Comunidad de Software Libre de la UCI como un entorno real para aplicar dos experimentos que prueban, que al utilizar el complemento implementado se minimiza el esfuerzo dedicado, y se elimina el efecto de la influencia de la subjetividad al analizar los comentarios de forma tradicional. Además de confrontar el placer de uso entre realizar las tareas de manera tradicional y usando el complemento implementado.

3.2 Aplicación de métricas de evaluación de rendimiento de algoritmos de agrupamiento

Para poder evaluar las soluciones de los distintos algoritmos de agrupamiento que se aplican, es necesario disponer de un conjunto de datos, sobre el cual se ejecutan estos algoritmos, y que cuentan con diferentes particularidades. Dentro de los variados tipos de conjuntos de datos que se pueden encontrar para las distintas actividades dentro del área del aprendizaje automático, para este estudio resultan de interés aquellos conjuntos de datos que se puedan agrupar relativamente bien según sus características para lograr soluciones que no carezcan de sentido y que resulten útiles para el cumplimiento de los objetivos planteados (Salinas Dezerega, 2016).

El conjunto de datos denominado **20Newsgroups** es una colección de aproximadamente 20,000 documentos de grupos de noticias, particionados casi homogéneamente a través de 20 distintos grupos de noticias. Los grupos de noticias son tipos de foros en línea en los que se puede discutir abiertamente sobre distintos temas de interés de las personas que participan, donde cada una puede escribir comentarios en el grupo de noticias adecuado. Esta colección de documentos ha sido un conjunto de datos popular para realizar experimentos en aplicaciones de texto en técnicas del aprendizaje automático, tales como clasificación de texto y agrupamiento de texto (Salinas Dezerega, 2016).

Este conjunto de datos ha sido popularmente usado en áreas del aprendizaje automático relacionadas con clasificación, agrupamiento y modelos de tópicos, por algoritmos tales como K-

[10] Blog de la comunidad de software libre de la UCI, disponible en: https://humanos.uci.cu/

means. Como este algoritmo es usado en este trabajo, debido a que se encuentra disponible en la biblioteca scikit-learn de Python, se utiliza el conjunto de datos de grupos de noticias debido a que es uno de los que se encuentra mejor documentado (Salinas Dezerega, 2016).

Las métricas que se emplean para evaluar los resultados obtenidos (ver epígrafe 2.6) tras la ejecución de los algoritmos designados, sobre los conjuntos de datos seleccionados, dependen del análisis que se quiera lograr, que, en este caso, incluye dos aspectos del proceso de experimentación: la eficiencia de los algoritmos que se ejecutan, y la calidad de las soluciones que éstos produzcan.

Para evaluar la eficiencia de los algoritmos, según la biblioteca en la que son implementados, se utiliza como métrica el tiempo total de ejecución de los algoritmos, haciendo una separación, en los casos en que fuera posible, entre el tiempo de lectura y el de procesamiento de los datos, para cada algoritmo.

Por otro lado, el rendimiento de los algoritmos es medida mediante varias funciones provistas por la biblioteca scikit-learn. Estas funciones son:

- Índice de Rand ajustado.
- Homogeneidad, integridad y V-measure.
- Coeficiente de la silueta.

3.2.1 Resultado de la aplicación de las métricas

Primero se carga el conjunto de datos 20Newsgroups del cual se definen 3529 elementos de muestra para analizar. Luego se procesan los textos y se construye un espacio con los vectores asignados a cada post. Luego se imprimen los resultados de aplicar las métricas (ver figura 12).

```
from sklearn import metrics

print 'Metricas'

print("Homogeneity: %0.3f" % metrics.homogeneity_score(labels, km.labels_))

print("Completeness: %0.3f" % metrics.completeness_score(labels, km.labels_))

print("V-measure: %0.3f" % metrics.v_measure_score(labels, km.labels_))

print("Adjusted Rand Index: %0.3f" %
 metrics.adjusted_rand_score(labels, km.labels_))

print(("Silhouette Coefficient: %0.3f" %
 metrics.silhouette_score(vectorized, labels, sample_size=1000)))
```

Figura 12 Evaluación del rendimiento del agrupamiento. (Elaboración propia)

Para los datos obtenidos en la aplicación de las métricas estudiadas se alcanzaron los siguientes resultados:

- Homogeneidad: se obtuvo un valor de 0.400.
- Integridad: se obtuvo un valor de 0.206.
- V-measure: se obtuvo un valor de 0.272.
- Índice de Rand ajustado: se obtuvo un valor de 0.064.
- Coeficiente de la silueta: se obtuvo un valor de 0.006.

Con estos resultados se estima el rendimiento del método propuesto como solución, a partir del conjunto de datos **20Newsgroups.** Los resultados en general fueron buenos, lo cual se puede apreciar a partir de los resultados obtenidos en cada una de las métricas.

Las medidas de Homogeneidad e Integridad (ver epígrafe 2.6.2), representan que los grupos solamente contengan miembros de una sola clase y que todos los miembros de una clase dada se asignen al mismo grupo, respectivamente, se limita a un intervalo de 0 a 1, siendo 1 el mejor resultado posible; significando, que el valor obtenido de 0.400 y de 0.206 respectivamente, y el valor de V-measure que es la media armónica entre estos valores, es bueno, dado el conjunto de datos y el número de grupos escogidos.

El índice de Rand ajustado (ver epígrafe 2.6.1), que es una función que mide la similitud entre lo: se obtuvo y el agrupamiento correcto, se limita a un intervalo de -1 a 1, siendo los valores negativos resultados malos (etiquetas independientes), por lo que, al resultar un valor de 0.064, se puede afirmar que los agrupamientos son similares, dado el conjunto de datos y el número de grupos escogidos.

El Coeficiente de la silueta (ver epígrafe 2.6.3), que mide que los grupos resultantes se encuentren bien definidos, se limita a un intervalo de -1 a 1, siendo los valores negativos resultados malos (agrupación incorrecta), por lo que, al resultar un valor de 0.006, se puede afirmar que la agrupación es relativamente densa, dado el conjunto de datos y el número de grupos escogidos.

Por los resultados obtenidos, se concluye que según todos los indicadores el algoritmo posee un buen rendimiento, lo que indica que es preciso¨

3.3 Aplicación en un entorno real

El Blog HumanOS es un espacio creado por estudiantes, profesores y trabajadores de la Universidad de las Ciencias Informáticas en el 2009. HumanOS surgió para contribuir al fomento del uso del software libre en la UCI. El propósito de este blog es mostrar a las personas cuáles son las ventajas del software libre, así como mantener a toda la comunidad informada acerca de la

actualidad de GNU/Linux. Además, cuenta con varios espacios de intercambio directo entre sus miembros (peñas, ferias expositivas, conferencias, talleres).

3.3.1 Obtención de los comentarios

A partir del artículo **#Aviso – Actualización #facebook – #pidgin**[11], el cual se refiere a una actualización de un complemento para utilizar el cliente de mensajería Pidgin para conectarse al chat de Facebook, se obtuvieron 20 comentarios en el momento en que se realizó la extracción de datos (ver anexo 2).

Figura 13 Extracción de los comentarios. (Elaboración propia)

[11] Artículo disponible en: http://humanos.uci.cu/2017/04/comosehace-regresando-facebook-al-pidgin/

3.3.2 Utilización de un experimento para la validación del método desarrollado

Un experimento es todo un proceso complejo en el que se emplean medidas y se realizan pruebas para comprobar y estudiar algún proceso antes de ejecutarlo por completo. En un experimento se realizan todo tipo de estudios, a fin de constatar la funcionalidad del objeto en estudio. Teorías e hipótesis nacen a partir de los experimentos que se realizan en torno a una premisa. Los experimentos son de vital importancia en el campo científico, y son parte esencial de los estudios que se realizan en un ambiente controlado.

El experimento científico es la evaluación práctica de las teorías que han surgido de una teoría, a partir de aquí se desarrollan hipótesis y más teorías que son valoradas de la misma manera generando una cadena experimental que culmina en la realización de una conclusión que arroja los resultados de todos los experimentos.

Por supuesto, es común asociar el término experimento al campo científico, pero en realidad los experimentos se utilizan para evaluar el comportamiento de otros agentes cuya forma o comportamiento importan para algún sector en específico. Tal es el caso de los estudios poblacionales; en ellos se hacen experimentos de comportamiento personal, donde que se ponen a prueba un conjunto determinado de personas, a las cuales se asigna una o varias tareas y se analiza sus respectivos comportamientos. Para garantizar los resultados obtenidos con la aplicación del método desarrollado, se propone un experimento en el que se comparen los resultados obtenidos de la aplicación de técnicas tradicionales de análisis de comentarios y los resultados de aplicar el método propuesto.

El objetivo de la aplicación de esta técnica en el presente trabajo es verificar la utilidad del complemento implementado, el cual minimiza el esfuerzo dedicado y elimina el efecto de la influencia de la subjetividad al analizar los comentarios de forma tradicional, además de medir la respuesta emocional de forma no verbal.

Subjetividad en la toma de decisiones

La toma de decisiones es un acto que implica tener que optar, enjuiciar y replantear las relaciones de las variables que conforman dicha situación, por lo que tienen un carácter complejo, y como todo elemento complejo tiene a su vez múltiples factores que determinan dicha acción. Los elementos distintivos dentro de esa acción se pueden clasificar en lo emocional y lo racional. Lo emocional representa lo subjetivo del ser, y lo racional está enmarcado en un espacio y tiempo determinado, donde se transcribe el sujeto y la influencia de la sociedad en su actuar. En este sentido se puede comprender que la toma de decisiones depende completamente de la persona en sí (Hampshire, y otros, 1958).

Los individuos eligen y deciden de acuerdo con los sistemas de creencias y prácticas imperantes en la sociedad, de tal manera, que su comportamiento es “consciente” y/o “coherente” según estos, por lo que el sujeto toma las decisiones bajo su posicionamiento, y según su habitus[12].

La toma de decisiones esta contrarrestada por dos “fuerzas”: una en el campo racional que induce al individuo a razonar, reflexionar y actuar en lo políticamente correcto, mientras que la otra fuerza impulsa lo que está dentro del ser: las emociones, pensamientos e ideas que van moldeando las facultades de este (Hampshire, y otros, 1958). Esto demuestra, que la toma de decisiones está influenciada por cada individuo, donde muchas veces lo emocional le gana a lo racional, por lo que el uso del complemento implementado elimina el efecto de la influencia de la subjetividad en el agrupamiento de comentarios de usuarios de manera tradicional.

3.3.2.1 Aplicación de los experimentos

Para la aplicación eficaz de las técnicas propuestas y la validez de los resultados se precisa disponer de una muestra confiable y representativa de usuarios reales o potenciales de la solución. La determinación de una muestra descarta criterios estadísticos para basarse en el estudio “Number of Participants in User Testing” de Nielsen (Nielsen, 2012), en que se proponen la selección de grupos entre 5 y 15 usuarios para la aplicación de las técnicas. Nielsen pudo demostrar con 83 casos de estudio que una cifra mayor a 15 usuarios en la muestra obtendría similares resultados cuando se aplican técnicas de usabilidad, y que, por tanto, en ese rango se reportaban la mayoría de los hallazgos. Partiendo de este aporte y tomando en consideración la naturaleza de las técnicas propuestas se decide que los grupos de prueba de la muestra no tengan un tamaño mayor a 15 (ver figura 14).

Sin embargo, en el estudio el valor fundamental de la muestra no descansa en su tamaño, sino en su representatividad. Para la eventual generalización de los resultados, la muestra debe representar la mayoría de las propiedades de la población. Teniendo en cuenta su caracterización (se decide que cada grupo de prueba se acerque al mismo cuadro de representación por estratos. En esta selección muestral resulta significativo el enfoque de género, el balance etnográfico territorial y racial, el estatus, el grupo etario-generacional y en el caso de los estudiantes, la evolución cognitiva en la carrera. El método de selección seleccionado es el aleatorio simple. Es importante señalar

[12] Habitus: Condicionamientos asociados a una clase particular de condiciones existentes. (concepto creado por Pierre Bourdieu)

que otros indicadores interesantes emergieron durante la intervención, como el credo, la afiliación política y la orientación sexual, lo cual impregna al estudio de un enfoque holístico.

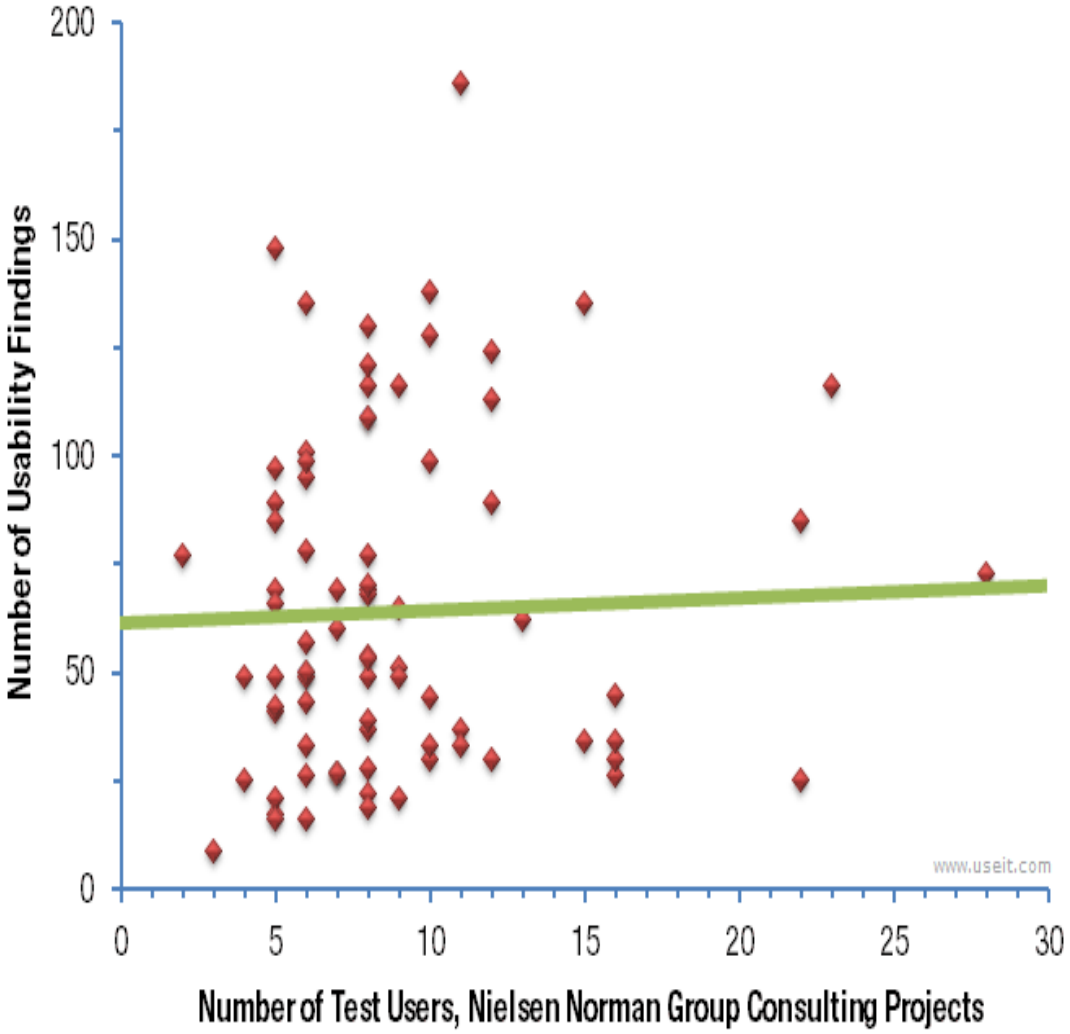

Figura 14 Estudio de Nielsen sobre la cantidad de usuarios para realizar pruebas de usabilidad (Nielsen, 2012)

Para la aplicación de los experimentos se tomaron los 20 comentarios obtenidos y dos muestras de 5 y 8 personas, respectivamente; y se les pide que realicen un grupo de tareas con el fin de obtener de la primera muestra (Grupo 1) una asignación de los comentarios dados cuatro grupos predefinidos, y de la segunda muestra (Grupo 2) que identifiquen la cantidad de grupos que cada persona cree que existe en estos comentarios. Además, se mide la respuesta emocional de forma no verbal a partir de una técnica efectiva para la mediación de la misma, la técnica Emocard.

Con el resultado de la aplicación de estos experimentos se espera comparar las elecciones de las agrupaciones obtenidas por cada grupo y el método desarrollado. Además de confrontar el placer

de uso entre realizar las tareas de manera tradicional y usando el complemento implementado a través del método Emocard.

Las tareas fueron elaboradas por un analista de datos de la Universidad de las Ciencias Informáticas.

3.3.3 Correlación entre los resultados del experimento y los resultados del método desarrollado

Se entiende como correlación al grado de relación existente entre dos variables. Para verificar la correspondencia de los resultados obtenidos con el experimento y con el método desarrollado se aplicaron dos pruebas. Se aplica también un diagrama de dispersión para poder observar de forma gráfica la dispersión de los datos

3.3.4 Diagrama de dispersión

Un diagrama de dispersión es una representación gráfica de la relación entre dos variables, muy utilizada en las fases de comprobación de teorías e identificación de causas raíz y en el diseño de soluciones y mantenimiento de los resultados obtenidos.

Estudios revelan que resulta más fácil identificar las relaciones en un diagrama de dispersión que en una tabla de números (Centro de excelencia, 2006).

El análisis de un diagrama de dispersión consta de un proceso de cuatro pasos: se elabora una teoría razonable, se obtienen los pares de valores y se dibuja el diagrama, se identifica la pauta de correlación y se estudian las posibles explicaciones. Las pautas de correlación más comunes son: correlación fuerte positiva (Y aumenta claramente con X), correlación fuerte negativa (Y disminuye claramente con X), correlación débil positiva (Y aumenta algo con X), correlación débil negativa (Y disminuye algo con X), correlación compleja (Y parece relacionarse con X pero no de un modo lineal) y correlación nula (no hay relación entre X e Y) (Centro de excelencia, 2006).

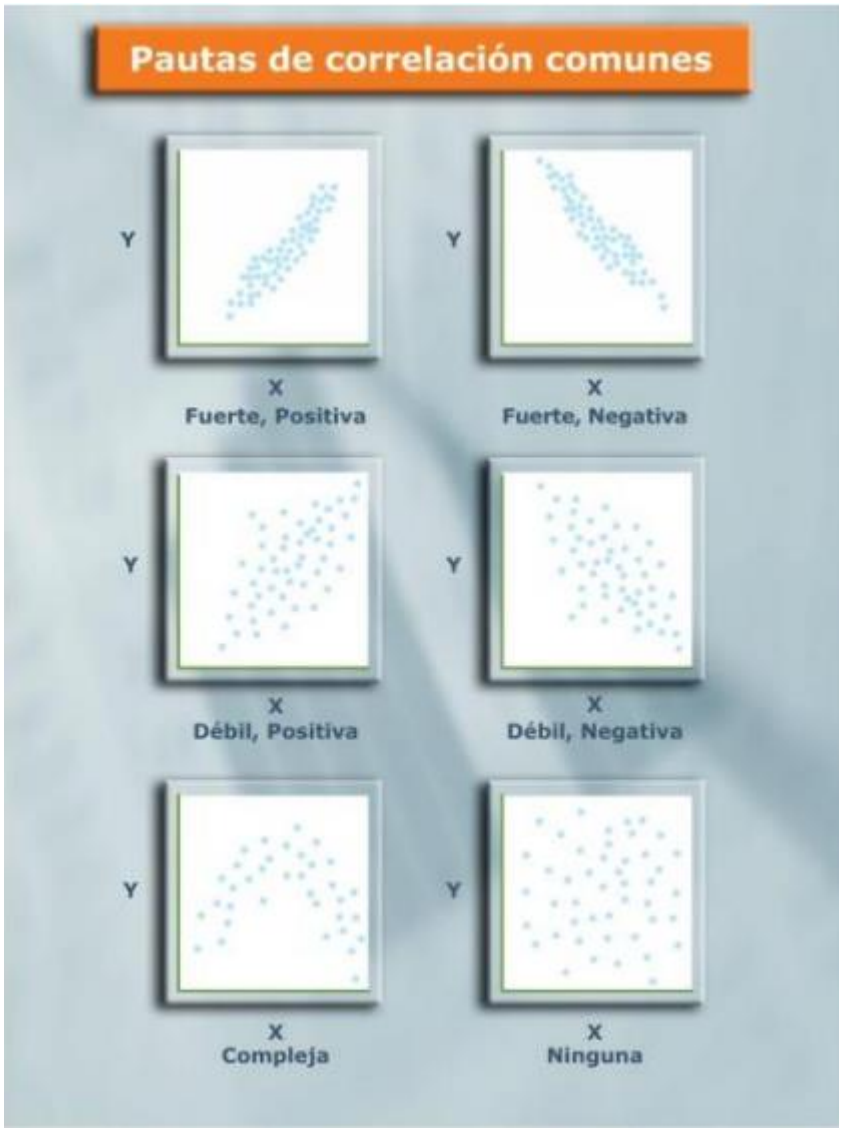

Figura 15: Diagrama de dispersión (Centro de excelencia, 2006)

3.3.5 Aplicación del coeficiente basado en la correlación de Spearman y Pearson

Para la aplicación del coeficiente se definieron las siguientes premisas: si se obtiene un valor de correspondencia mayor o igual que 0,6 y menor que 0,8, entonces el valor de correspondencia es adecuado; si se obtiene un valor de correspondencia mayor o igual que 0,8, entonces el valor de correspondencia es muy confiable.

0,6 <= x < 0,8 Adecuado

x >= 0,8 Muy confiable

ID Comentario	Sujeto A	Sujeto B	Sujeto C	Sujeto D	Sujeto E
1	1	1	1	1	1
2	1	1	1	0	1
3	1	0	1	1	1
4	1	0	1	1	1
5	1	1	1	1	1
6	1	1	1	1	1
7	1	1	0	1	1
8	1	1	1	1	0
9	1	1	1	1	1
10	0	1	1	1	1
11	0	1	1	1	1
12	1	1	1	0	1
13	1	1	0	1	1
14	1	1	1	1	1
15	0	1	1	1	1
16	1	1	1	1	1
17	1	1	1	1	1
18	1	1	1	1	0
19	1	1	1	1	0
20	1	1	0	1	1
Correlación resultante:	**0.87**				

Figura 16 Tabla del resultado de la aplicación del experimento en el Grupo 1. (Elaboración propia)

Cantidad de grupos identificados	Sujeto A	Sujeto B	Sujeto C	Sujeto D	Sujeto E	Sujeto F	Sujeto G	Sujeto H
1								
2								
3								
4	1		1	1	1	1	1	1
5		1						
6								

	Sujeto A	Sujeto B	Sujeto C	Sujeto D	Sujeto E	Sujeto F	Sujeto G	Sujeto H
Identificó correctamente	1	0	1	1	1	1	1	1

Grado de correlación	0.875

Figura 17 Tabla del resultado de la aplicación del experimento en el Grupo 2. (Elaboración propia)

Tiempo de procesamiento

Durante la realización de los experimentos se tuvo en cuenta el tiempo en que los sujetos de ambos grupos demoraron en la realización de las tareas. Este tiempo se compara con el tiempo de ejecución del complemento para el procesamiento de los comentarios. Las figuras 18 y 19 muestran la diferencia entre dichos tiempos.

Sujetos	A	B	C	D	E	Tiempo promedio	Complemento	Variación del tiempo
Tiempo en minutos	20	25	19	26	21	22,2	0,2	22

Figura 18 Tiempo de realización de las tareas del Grupo 1 y del complemento implementado.

Sujetos	A	B	C	D	E	F	G	H	Tiempo promedio	Complemento	Variación del tiempo
Tiempo en minutos	15	14	20	22	18	23	14	15	17,625	0,2	17,425

Figura 19 Tiempo de realización de las tareas del Grupo 2 y del complemento implementado.

El grado de correlación resultante luego de aplicar dicho coeficiente a los resultados del Grupo 1 es de 0.870, y del Grupo 2 de 0.875, por lo que se puede verificar que la correlación existente es una correlación muy confiable; lo que prueba, que al utilizar el complemento implementado se minimiza el esfuerzo dedicado, se elimina el efecto de la influencia de la subjetividad y se disminuye el tiempo necesario para el procesamiento de los comentarios de usuarios.

Diagrama de dispersión

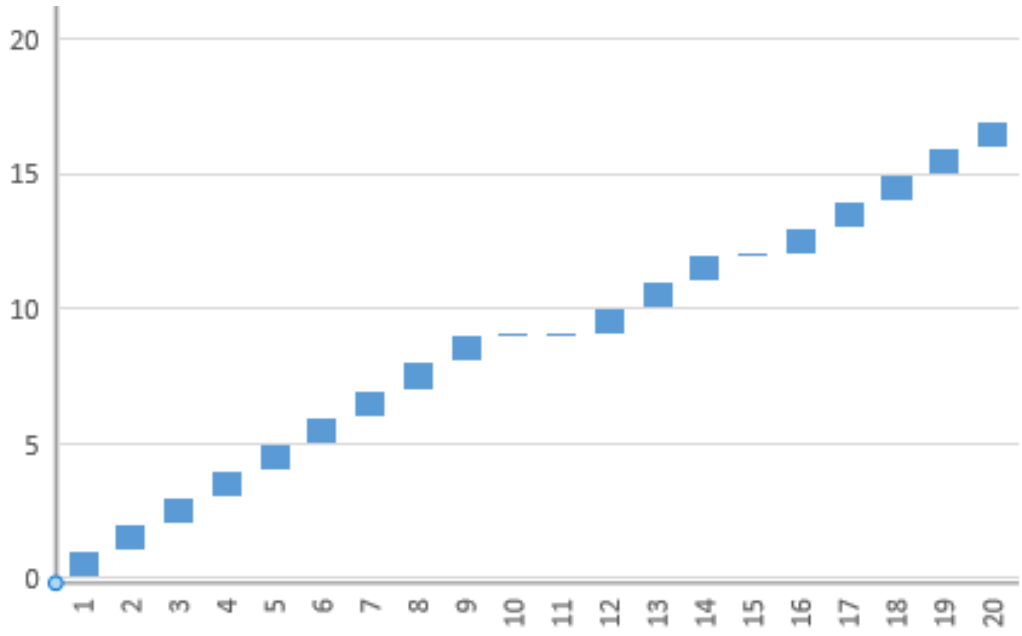

Figura 20 Diagrama de dispersión experimento / complemento (Elaboración propia)

Al observar la gráfica se puede concluir que la dispersión obtenida presenta una correlación fuerte positiva (los valores de Y aumentan con los de X).

3.4 Aplicación de la técnica Emocard

Es una técnica efectiva para la mediación de la respuesta emocional de forma no verbal. Utiliza tarjetas que permiten a los usuarios expresar lo que sienten en el instante de forma sencilla,

eliminando las limitaciones en el lenguaje. Para los usuarios supone una dificultad expresar con palabras sus emociones, y es común que los evaluadores no logren encontrar realmente la verdadera sensación que provoca la aplicación. Esta técnica permite al usuario identificar (a través de rostros que emulan estados de ánimos) cómo se siente al realizar el agrupamiento de la manera tradicional y al interactuar con el sistema.

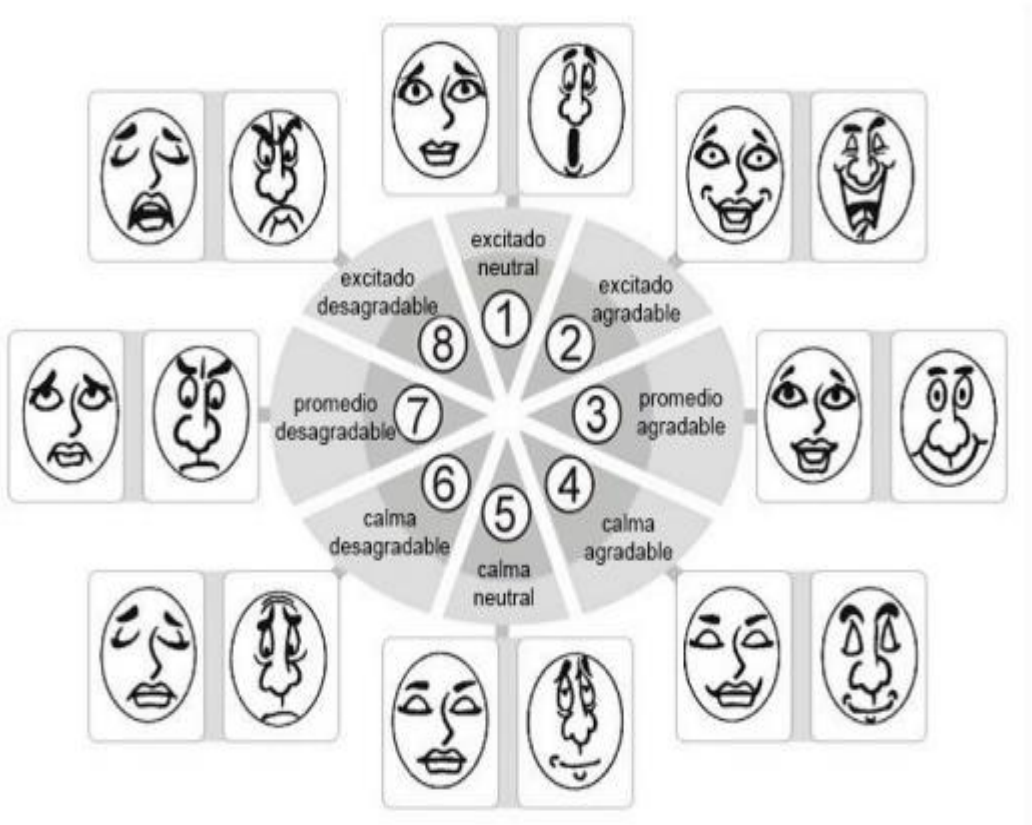

Figura 21: Emocard (Agarwal, et al., 2009)

Se evalúa una muestra de 6 sujetos (el tamaño de la muestra ha sido determinado según el informe “Beyond Usability” (Agarwal, et al., 2009) seleccionando tres de cada una de las muestras de los experimentos anteriores, a los que se le entregan, según su género (masculino o femenino) un juego de 8 tarjetas con rostros de hombre o mujer, en donde están registrados diversos estados desde “excitación” hasta “calma” y con inclinaciones negativas o positivas (Figura 21).

Para el completamiento de esta prueba a los usuarios se les pide reflejar su estado de ánimo en dos momentos diferentes, al procesar los comentarios de manera tradicional y al realizar esta tarea utilizando el complemento implementado. Los dos grupos de resultados quedan de la siguiente manera:

Primer momento: Respuesta emocional del usuario al procesar los comentarios de manera tradicional.

Segundo momento: Respuesta emocional del usuario al procesar los comentarios mediante el complemento implementado.

Una vez que los usuarios concluyen la realización de ambas tareas se les pide que reflejen sus emociones mediante la técnica de Emocard, resultando de la siguiente forma:

Forma tradicional	
Usuario	Elección Emocard
a	1
b	6
c	8
d	5
e	6
f	4

Complemento	
Usuario	Elección Emocard
a	2
b	3
c	2
d	3
e	1
f	2

Figura 22: Resultado de aplicar técnica Emocard. (Elaboración propia)

Traduciendo esta tabla en una imagen de estudio de la técnica quedaría de la siguiente manera (Figura 23):

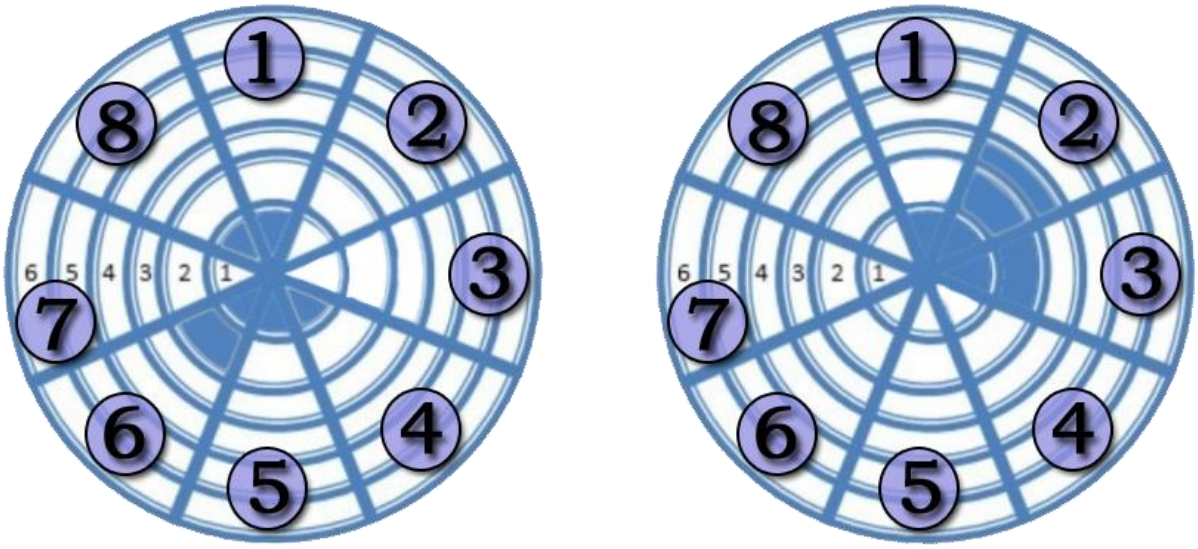

Figura 23 Resultados de Emocard. (Elaboración propia)

En la representación gráfica se evidencia como con el uso del complemento los usuarios coinciden entre la categoría de agradable y neutral. No sucede lo mismo con la forma tradicional pues esta logra una menor aceptación, los resultados muestran una mayor incidencia en la categoría neutral con una tendencia hacia la no aceptación.

Por tanto, se puede afirmar por los resultados obtenidos que, los usuarios al realizar la tarea de procesamiento mediante el uso del complemento, reaccionan de una forma más agradable y con mayor nivel de aceptación, a lo experimentado mediante el procesamiento de forma tradicional.

3.5 Conclusiones del capítulo

En este capítulo se han presentados un conjunto de técnicas en calidad de prueba que permitieron contrastar empíricamente los resultados obtenidos y validar su efectividad. Estas pruebas han corroborado la precisión de la solución implementada y demostraron su utilidad y placer de uso.

La aplicación de mediciones basadas en el coeficiente de correlación de Spearman y Pearson verificó la correspondencia entre los datos obtenidos en los experimentos y aquellos obtenidos de la solución implementada, lo cual demostró además que los niveles de correlación aumentan en la medida que se incorporan mayor cantidad de datos, elevando la precisión de los análisis y con ello la fiabilidad de los resultados obtenidos.

CONCLUSIONES GENERALES

En función de los objetivos de la investigación los resultados permitieron arribar a las siguientes conclusiones:

- El análisis del proceso de descubrimiento de conocimiento en bases de datos y los enfoques utilizados en técnicas de agrupamiento en la minería de texto permitieron identificar características competitivas y tecnologías para definir e implementar un método de agrupamiento de comentarios de usuarios.
- Se obtuvo un complemento implementado en Python para el sistema AOpinion que aplicando el método de agrupamiento propuesto facilitó la detección de información relevante durante el procesamiento de comentarios de usuarios en las aplicaciones de software, lo cual promovió el estudio y la comprensión del comportamiento de los comentarios de usuarios acerca de las aplicaciones de software.
- La aplicación de mediciones basadas en el coeficiente estadístico de correlación de Spearman y Pearson verificaron la correspondencia entre los datos obtenidos por el complemento implementado y aquellos aportados en los experimentos de validación, lo cual demostró además que los niveles de correlación aumentan en la medida que se incrementa el volumen de información, elevando la precisión de los análisis y con ello la fiabilidad de los datos, lo cual demostró la efectividad del método de agrupamiento obtenido al minimizar el esfuerzo dedicado, eliminar el efecto de influencia de la subjetividad y disminuir considerablemente el tiempo necesario para el procesamiento de los comentarios de usuarios.
- Los resultados obtenidos demostraron que el desarrollo del método de agrupamiento propuesto eleva la precisión con que se obtiene la información relevante de los usuarios sobre las aplicaciones de software y demuestra su potencial de generalización en otros entornos, lo cual representa una contribución sustancial al proceso de mejora y evolución de las aplicaciones en la UCI.

RECOMENDACIONES

- Incluir en la aplicación AOpinion la posibilidad de buscar y recopilar información sobre una aplicación específica en redes sociales como Twitter o Facebook.
- Agregarle a la aplicación AOpinion la posibilidad de procesar los comentarios de usuarios desde archivos en diferentes formatos (por ejemplo, archivos xml o csv).
- Añadir el análisis de influencia entre los autores de los comentarios.

BIBLIOGRAFÍA

Agarwal, Anshu y Meyer, Andrew. 2009. *Beyond Usability: Evaluating Emotional Response as an Integral Part of the User Experience.* Boston, MA, USA : En CHI'09 Extended Abstracts on Human Factors in Computing Systems, 2009.

Allic, Mickel. 2010. The Importance of User Feedback. *PixelTango.* [En línea] 2010. [Citado el: 04 de 04 de 2016.] https://www.pixeltango.com/articles/interaction-design/the-importance-of-user-feedback/.

Álvarez, Miguel Ángel. 2009. Desarrolloweb.com. *Desarrolloweb.com.* [En línea] 14 de 10 de 2009. [Citado el: 03 de 04 de 2016.] http://www.desarrolloweb.com/articulos/que-es-html5.html.

Anton, César. 2015. Platzi. *Platzi.* [En línea] 09 de 07 de 2015. [Citado el: 06 de 04 de 2016.] https://platzi.com/blog/que-es-postgresql/.

Aplicaciones del procesamiento del lenguaje natural en la recuperación de información en español. **Vilares Ferro, Jesús. 2005.** 36, A Curuña : s.n., 2005. ISSN 1135-5948.

Azevedo, A y Santos, F. 2008. *SEMMA and CRISP-DM: a parallel overview. In Proceedings of the IADIS.* s.l. : European Conference on Data Mining, 2008.

Benavides Cañón, Paula Andrea y Rodríguez Correo, Sandra. *Procesamiento del lenguaje natural en la recuperación de información.* Bogotá : s.n.

Berry, Michael W. y Browne, Murray. 2005. *Understanding Search Engines: Mathematical Modeling and Text Retrieval.* s.l. : SIAM, Society for Industrial and Applied Mathematics, 2005.

Blanco-Hermida Sanz, Eric Joel. 2016. *Algoritmos de clustering y aprendizaje automático aplicados a Twitter.* Cataluña : s.n., 2016.

Centro de excelencia. 2006. Diagrama de dispersión. [En línea] 2006. [Citado el: 05 de 05 de 2014.] http://www.centrosdeexcelencia.com/entidades/herram/dispersion.

Chapman, R, y otros. 2000. *CRISP-DM 1.0 Step-by-step data mining guide.* s.l. : The CRISP-DM consortium, 2000.

Chen, Edwin. 2016. Introduction to Latent Dirichlet Allocation. [En línea] 2016. http://blog.echen.me/2011/08/22/introduction-to-latent-dirichlet-allocation/.

Crider, Michael. 2014. Narcissistic Jerks Are Giving Play Store Apps 1-Star Reviews For Higher Visibility. *Android Police.* [En línea] 12 de 01 de 2014. [Citado el: 18 de 01 de 2016.]

http://www.androidpolice.com/2014/01/12/narcissistic-jerks-are-giving-play-store-apps-1-star-reviews-for-higher-visibility/.

Django. 2016. Django en español. *Django en español.* [En línea] 2016. [Citado el: 01 de 03 de 2016.] http://django.es/.

Feldman, Ronen y Dagan, Ido. 1995. *Knowledge Discovery in Textual Databases.* Ramat Gan : s.n., 1995.

Feldman, Ronen y Sanger, James. 2007. *The Text Mining Handbook.* Cambridge : s.n., 2007.

Fontela, Alvaro. Raiola Networks. *Raiola Networks.* [En línea] [Citado el: 05 de 05 de 2016.] https://raiolanetworks.es/blog/que-es-bootstrap/.

Frain, Ben. 2012. *Responsive web design with HTML5 and CSS3.* s.l. : Packt Publishing Ltd, 2012.

Gallardo San Salvador, José Ángel. 2015. Programa de la asignatura Ampliación de Análisis de Datos Multivariantes. *Universidad de Granada.* [En línea] Departamento de Estadística e Investigación Operativa, 2015. http://www.ugr.es/.

González Duque, Raúl. 2015. *Python para todos.* 2015.

Guevara López, Rubén. 2011. *Minería de textos en la red social Twitter.* s.l. : FACULTAD DE INGENIERÍA, UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO, 2011.

Hampshire, Stuart y Hart, Herbert. 1958. *Decision, intention and certainty.* L.A. : Mind, 1958.

Hernández Orallo, José, Ferri Ramírez, César y Ramírez Quintana, M. José. 2004. *Introducción a la Minería de Datos.* Madrid : Pearson Educación, 2004. 28042.

Hernández, Francisco Refugio Zavala. 2014. *Buscador de artículos científicos aplicando Minería de datos.* México : s.n., 2014.

Herramienta informática de Minería de Uso de la Web sobre los registros de navegación por Internet. **Ordoñes Leyva, Yoanni y Avilés Vázquez, Ernesto. 2010.** La Habana : s.n., 2010.

Huang, Anna. 2008. *Similarity Measures for Text Document Clustering.* s.l. : Depar tment of Computer Science, The University of Waikato, Hamilton, New Zealand, 2008.

Jojo. 2011. How important is feedback? *UX User Experience StackExchange.* [En línea] 30 de 07 de 2011. [Citado el: 19 de 01 de 2016.] http://ux.stackexchange.com/questions/9472/how-important-is-feedback.

Kanungo, Tapas. 2002. *An efficient k-means clustering algorithm: Analysis and implementation.* s.l. : IEEE transactions on pattern analysis and machine intelligence, 2002.

Larsen, Bjornar y Aone, Chinatsu. 1999. *Fast and effective text mining using linear-time document clustering.* s.l. : Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining, 1999.

Martínez, Rafael. 2013. PostgreSQL-es. *PostgreSQL-es.* [En línea] 2013. [Citado el: 19 de 01 de 2016.] http://www.postgresql.org.es/sobre_postgresql.

matplotlib.org. 2015. matplotlib.org. [En línea] 2015. http://matplotlib.org.

Mendoza, Milagros. 2014. Tienda Nube. *Tienda Nube.* [En línea] 23 de 06 de 2014. [Citado el: 15 de 04 de 2016.] https://www.tiendanube.com/blog/herramientas-y-consejos-de-monitoreo-redes-sociales/.

Minería de Datos como soporte a la toma de decisiones empresarial. **Marcano Aular, Yelitza Josefina y Talavera Pereira, Rosalba. 2007.** 52, 2007.

Moine, Juan Miguel. 2013. Metodologías para el descubrimiento de conocimiento en bases de datos: un estudio comparativo. 2013.

Murtagh, Fionn y Legendre, Pierre. 2014. *Ward's hierarchical agglomerative clustering method: which algorithms implement ward's criterion?* s.l. : Journal of Classification, 2014.

Nielsen, Jakob. 2012. Alertbox: Number of Participants in User Testing. [En línea] 2012. http://www.useit.com/alertbox/number-of-test-users.html.

Olson, David L y Delen, Dursun. 2008. *Advanced Data Mining Techniques.* Berlin, Germany : Springer publishing, 2008.

Pérez Valdés, Damián. 2007. MAESTROS DEL WEB. *MAESTROS DEL WEB.* [En línea] 03 de 07 de 2007. [Citado el: 03 de 04 de 2016.] http://www.maestrosdelweb.com/que-es-javascript/.

Pérez, Lorenzo Villarroel. 2015. *"Mining Mobile Apps Reviews to Support Release Planning".* Madrid : U N I V E R S I D A D P O L I T É C N I C A, 2015.

Polettini, Nicola. 2004. *"The Vector Space Model in Information Retrieval- Term Weighting Problem".* s.l. : Department of Information and Communication Technology, Universidad de Trento, 2004.

PosNeg opinion: Una herramienta para gestionar comentarios de la web. **Amores, Mario, Arco, Leticia y Artiles, Michel. 2015.** 1, Villa Clara : s.n., 2015, Vol. 9.

Probabilistic Topic Models. **Blei, David. 2012.** s.l. : Communications of the ACM, 2012.

Requests. 2014. Requests. [En línea] 2014. https://pypi.python.org/pypi/requests.

Richardson, Leonard. 2012. BeautifulSoup. *crummy.com.* [En línea] 2012. [Citado el: 30 de 5 de 2017.] https://www.crummy.com/software/BeautifulSoup/.

Salinas Dezerega, Francisco Javier. 2016. *"Evaluación de algoritmos de agrupamiento utilizando apache spark".* Santiago de Chile : Departamento De Informática, Universidad Técnica Federico Santa María, 2016.

Scikit-learn. 2017. Scikit-learn website. *Sklearn.* [En línea] 2017. http://scikit-learn.org/stable/modules/clustering.html.

Steinbach, Michael. 2000. *A comparison of document clustering techniques.* s.l. : KDD workshop on text mining., 2000.

Universidad de las Ciencias Informáticas. 2016. *Informe de la Infraestructura Productiva.* Dirección General de Producción. 2016. sn.

Viera, Ángel Freddy Godoy. 2017. *Técnicas de aprendizaje de máquina utilizadas para la minería de texto.* s.l. : Investigación Bibliotecológica. Archivonomía, Bibliotecología e Información, 2017.

Xu, Wei, Liu, Xin y Gong, Yihong. 2003. *Document clustering based on non-negative matrix factorization.* s.l. : Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval, 2003.