UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE INGENIERÍA INDUSTRIAL

EXTRACCIÓN DE INFORMACIÓN Y CONOCIMIENTO DE LAS OPINIONES EMITIDAS POR LOS USUARIOS DE LOS SISTEMAS WEB 2.0

TESIS PARA OPTAR AL GRADO DE MAGÍSTER EN GESTIÓN DE OPERACIONES

MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL INDUSTRIAL

RODRIGO ALFONSO DUEÑAS FERNÁNDEZ

PROFESOR GUÍA:
JUAN D. VELÁSQUEZ SILVA

MIEMBROS DE LA COMISIÓN:
PABLO E. ROMÁN ASENJO
ÁNGEL JIMENEZ MOLINA
GASTÓN L'HUILLIER CHAPARRO

SANTIAGO DE CHILE
SEPTIEMBRE 2013

Resumen

El objetivo de este trabajo de tesis es desarrollar una plataforma informática *Web Opinion Mining* (WOM) para la extracción de información que permita caracterizar la demanda de productos y servicios de una empresa, a través del uso de documentos publicados en sitios de noticias y las opiniones consignadas por los usuarios de las redes sociales.

En la sociedad de hoy, gracias a la aparición de la Web, el panorama competitivo de las empresas se ha vuelto mucho más complejo, debido a la cantidad de mercados interconectados en tiempo real que tienen que considerar. Por lo tanto, para obtener rendimientos sobre el promedio en este medio, es necesario tener nuevas maneras de predecir las acciones de la competencia y la demanda por productos y servicios.

Debido a lo anterior, la necesidad de procesar grandes cantidades de datos para obtener información ha ido creciendo a lo largo de los años. A medida que la capacidad de una empresa de procesar los datos de su entorno aumenta y se vuelve capaz de tomar decisiones estratégicas en base a la información obtenida, obtiene ventajas competitivas que reditúan en rendimientos sobre el promedio.

En base a lo anterior, es que se plantea la siguiente hipótesis de investigación: *"Las opiniones de los usuarios sobre productos y servicios de un nicho de mercado particular consignadas en los sistemas Web 2.0, contienen la información necesaria y suficiente para caracterizar su demanda aproximada"*.

Para probar esta hipótesis, se desarrollo una plataforma de detección de tendencias compuesta de tres módulos: minado de tópicos, minado de opiniones y visualización de tendencias. Debido a que esta plataforma tiene como objetivo apoyar la toma de decisiones de un grupo de expertos, es que se realizará el minado a partir de un conjunto predefinido de fuentes que describa el mercado que se quiere analizar.

El primer módulo se encarga de minar documentos de noticias y extraer qué tópicos están siendo publicados por semana. Una vez los tópicos de una semana son inferidos a través del uso del modelo de tópicos LDA, se generan *queries* para recuperar documentos opinados desde la red social *Twitter*, y se obtiene un puntaje de opinión para el tópico en particular durante esa semana. Una vez que se tiene información sobre un tópico por varios periodos, el módulo de visualización se encarga de entregar una representación gráfica de la evolución del tópico a lo largo del tiempo, tanto en documentos publicados como la opinión consignada por los usuarios en la Web 2.0.

Los resultados obtenidos por el módulo de minado de tópicos fue un precisión de 0.56, un recall de 0.52 y un F-Measure de 0.54. En el minado de opiniones se obtuvo un precisión de 0.6, 0.53 y 0.61 y un recall de 0.59, 0.49 y 0.58 para las polaridades positiva, neutra y negativa respectivamente. En el caso del modelo de tendencias, a medida que el umbral mínimo para considerar un evento como significativo aumenta, su precisión aumenta, llegando a una precisión de 0.61, recall de 0.51 y un F-Measure de 0.56.

Se concluye que el sistema propuesto para la representación de tendencias en la Web es un enfoque factible para modelar tendencias en la Web a través de la interacción de eventos, tópicos y opiniones consignadas en la Web 2.0. Por otro lado, los experimentos realizados comprueban la hipótesis planteada al inicio de este trabajo, ya que una vez toda la información es recolectada y analizada, es posible analizar el comportamiento de los tópicos a lo largo del tiempo, y ver como reaccionan los usuarios de la Web 2.0 y de manera indirecta caracterizar la demanda sobre ciertos productos y servicios.

Tabla de Contenido

Índice de figuras

Lista de Algoritmos

Capítulo 1

Introducción

En la actualidad la Internet y la World Wide Web se han convertido en piezas fundamentales del funcionamiento de nuestra sociedad, debido a la capacidad que otorgan de enlazar sistemas y el gran impacto que han tenido en las comunicaciones, las cuales son la base para lograr un mundo conectado donde no existen las fronteras a la hora de comunicarse. Esto ha tenido un gran impacto en las industrias, permitiendo que ellas se conecten entre si, y puedan utilizar una infinidad de servicios en tiempo real para todo tipo de aplicaciones.

La Web no sólo ha tenido injerencia en el desarrollo de la sociedad como un todo y del mundo empresarial, ya que con la paulatina llegada de la llamada Web 2.0, sus usuarios se vieron enfrentados a la posibilidad de utilizar este nuevo canal de comunicación como una herramienta no sólo para consumir conocimiento, si no también para comunicarse, publicar contenido y, gracias a la aparición de las redes sociales, expresar su opinión frente a un sinfín de temas. En esta misma línea, la evolución de la web ha traído consigo nuevas entidades tales como *wikis*, que permiten que los usuarios contribuyan con el fin de crear compendios de conocimiento y los *blogs*, que le dan la capacidad al usuario de publicar contenido sin que este pueda ser editados por otros [50].

Al cambiar a este nuevo paradigma, la Web se volvió una entidad en constante cambio a una tasa exponencial, ya que este nuevo paradigma permite que sus usuarios contribuyan con información y opinen sobre la variedad de temas que acontecen al día a día. Es esta última faceta la que conlleva una gran potencialidad, ya que a través de la expresión de sus opiniones, dan a conocer sus sentimientos y percepciones respecto de una variedad de temas, lo que permite conocer la percepción de un subconjunto de la sociedad sobre una variedad de temas, productos y servicios entre otros.

Ante esta nueva necesidad de análisis de datos, se han desarrollado una serie de herramientas y algoritmos para el procesamiento automático de las opiniones que los usuarios consignan en ambientes Web 2.0, las cuales se agrupan bajo el concepto de *Web Opinion Mining* [55]. Adicionalmente, se ha acuñado el término *Sentiment Analysis* [40], para denotar todo algoritmo o metodología que

busca analizar los sentimientos que guían a quien emite una opinión [39], con miras a determinar la sensibilidad y la relevancia con que los usuarios tratan prácticamente cualquier tema bajo estudio.

Uno de los temas que actualmente capta más miradas por parte de la comunidad científica, debido a su alto potencial económico, es el análisis de las opiniones y calificaciones sobre productos y servicios [38, 48], ya que si se realiza un consenso de la información disponible en la Web (tanto opiniones como contenido), es posible extraer conocimiento relevante sobre el valor que los usuarios otorgan a cada uno de ellos [2, 28, 71], y eventualmente obtener un panorama aproximado de las tendencias de los distintos nichos de mercado [12, 33].

Es por esto que esta tesis busca analizar el contenido textual presente en la Web, complementándolo con un análisis de las opiniones presentes en esta, para determinar las tendencias de algún nicho de mercado en particular. Se cree que será posible caracterizar la demanda futura por ciertos productos o servicios, a través de una correcta interpretación de la información extraída al aplicar un modelo de *análisis de tópicos* y *Web Opinion Mining* sobre los datos obtenidos de la Web o, en otras palabras, se busca detectar tendencias en un nicho de mercado en particular a través de la información pública presente en la web.

1.1. Planteamiento del Problema y Motivación

A medida que la tecnología avanza y se abre paso a un ritmo cada vez más vertiginoso al mundo empresarial, el panorama competitivo al que se enfrenta una empresa del siglo XXI se convierte en algo más que un conjunto de mercados independientes, los cuales antiguamente estaban tan sólo conectados por tratados de libre comercio y pactos de cooperación económica entre otros. El panorama de la competitividad empresarial en el mundo de hoy es el de un mundo globalizado, donde cualquier empresa de cualquier país puede incursionar en el mercado que más le acomode sin importar donde este se encuentre, o en varios mercados de manera simultánea si es que las sinergías operacionales que este posee así lo permiten.

Bajo este panorama competitivo de mercado global, es cada vez más necesario ser capaz de manejar grandes volúmenes de datos para gestionar de la mejor manera posible los recursos que se disponen, y al mismo tiempo, anticiparse a cada movimiento que realizará la competencia en busca de obtener ventajas competitivas, o impedir que otros las obtengan, para ser líderes en el mercado. El primer problema al que se debe enfrentar una empresa sumergida en el mundo globalizado, es el más complejo desde el punto de vista de la gestión de operaciones, por lo que varias metodologías y herramientas han nacido proponiendo soluciones, entre las cuales se encuentran los *Data Warehouses*, la *Business Intelligence* y el recientemente acuñado término de *BigData*. El segundo problema no

sólo involucra a la gestión de operaciones, ya que es necesario tener un equipo multidisciplinario encargado constantemente de monitorear el mercado, las acciones de las otras empresas, los anuncios presentes en los medios y cualquier indicio que permita anticiparse a los lanzamientos de productos y servicios de la competencia.

Una posible solución al problema planteado es minar la web en busca de esos indicios de manera automática, con foco en qué tópicos se habla en la Web, y analizando las redes sociales para estimar que percepción poseen los cibernautas sobre ellos. Es factible plantear la hipótesis de que a través de realizar un análisis de gran parte del conocimiento objetivo generado por los usuarios y los medios se puede atisbar aquellos indicios claves a la hora de plantear una planificación estratégica y operacional. Un sistema capaz de realizar esto de manera aproximada es realizable utilizando técnicas de recuperación de la información, modelamiento de tópicos y minado de opiniones sobre fuentes cuidadosamente seleccionadas que sean capaces de otorgarle al sistema una muestra significativa de todo lo que se habla sobre los mercados en los cuales se ve inmersa la empresa a nivel competitivo.

La motivación de esta tesis es realizar un primer acercamiento a la creación de semejante herramienta que se adapte a la ya mencionada metodología de trabajo, capaz de resolver de manera aproximada el problema de detección de tendencias en un mercado en particular. Si bien esta investigación busca resolver el problema de detección de tendencias, esta se enfocará en en análisis de tópicos y opiniones en la Web y modelar sus cambios a lo largo del tiempo.

1.2. Hipótesis de investigación

"Las opiniones de los usuarios sobre productos y servicios de un nicho de mercado particular consignadas en los sistemas Web 2.0, contienen la información necesaria y suficiente para caracterizar su demanda aproximada."

1.3. Objetivos

A continuación se presentan los objetivos de esta tesis, dando a conocer primero el objetivo general para luego dar un plan detallado de trabajo a través de sus objetivos específicos.

1.3.1. Objetivo General

"Diseñar y construir un sistema apoyado por Web Opinion Mining, aplicada en los datos que originan los usuarios de la Web 2.0, para caracterizar las tendencias de productos y servicios sobre un nicho de mercado en particular"

1.3.2. Objetivos Específicos

- Establecer un estado del arte sobre los algoritmos de *Web Opinion Mining* y extracción de tópicos desde documentos.
- Caracterizar las aplicaciones de estos algoritmos para la extracción de conocimiento.
- Desarrollar e implementar un algoritmo de recuperación de información desde sistemas Web 2.0 que permita asociar cada documento recuperado, sea opinado o no, a un punto específico en el tiempo.
- Diseñar e implementar un algoritmo de extracción de tópicos temporales desde una colección de documentos obtenida previamente desde la Web.
- Diseñar e implementar un modelo de *Web Opinion Mining*, junto con una función de score, que permita asignar un puntaje de opinión a cada tópico por unidad de tiempo.
- Diseñar un algoritmo que permit analizar la generación de ruido (o presencia) en las redes sociales a cada tópico por unidad de tiempo.
- Realizar una aplicación práctica de *Web Opinion Mining* junto con extracción de tópicos.
- Explicar y concluir sobre los resultados obtenidos.

1.4. Metodología

La metodología que se utilizará para llevar a cabo cada uno de los objetivos propuestos en esta investigación está compuesta por una serie de pasos en las que se entrelazan la extracción de tópicos desde documentos encontrados en la web y la extracción de opiniones asociadas a estos. Por lo tanto, los pasos a seguir en este trabajo serán los siguientes:

1. **Investigación de trabajo relacionado**

   Se realizará un estudio acabado de documentos pertinentes a los tópicos que serán mencionados en este trabajo. Principalmente se sobre enfocará en caracterizar el estado del arte sobre algoritmos y estrategias utilizadas en recuperación de la información, extracción de conocimiento (en particular tópicos presentes en documentos) y finalmente el minado de opiniones.

2. **Exploración del entorno actual en la Web 2.0**

   Se estudiarán los distintos sistemas Web 2.0 relevantes para este trabajo, entre los cuáles se considerarán blogs, sitios de *microblogging* y redes sociales. Se analizará su factibilidad para

ser minados junto con sus principales características, de manera de lograr un entendimiento acabado de la situación actual de la Web 2.0 a nivel mundial.

3. **Extracción de tópicos desde fuentes de contenido**

   Se desarrollará un modelo de extracción de tópicos desde fuentes de contenido a través de periodos temporales. Este modelo buscará implementar técnicas de recuperación de la información y de extracción de tópicos, que permitan caracterizar la evolución a través del tiempo de los tópicos presentes dentro del contenido recopilado de las fuentes seleccionadas.

4. **Análisis de las opiniones vertidas en los sistemas Web 2.0**

   Utilizando técnicas de *Web Opinion Mining*, se diseñará un modelo para analizar la evolución en el tiempo de la percepción por parte de los usuarios de los sistemas Web 2.0 previamente escogidos, sobre los tópicos que son extraídos de las fuentes de contenido.

5. **Aplicación del Modelo Propuesto**

   Se aplicará el modelo propuesto para tratar de detectar tendencias en productos y servicios en el nicho de la tecnología a nivel mundial, ya que actualmente es uno de los que más contenido y opiniones genera en la Web.

6. **Análisis de Resultados Obtenidos**

   Utilizando todo lo desarrollado en los pasos anteriores, se realizará un análisis de los resultados obtenidos, comparando la información recopilada con lo que comentan los expertos en la Web y con juicio de investigadores expertos.

7. **Conclusiones**

   Finalmente, se concluirá a partir de los resultados obtenidos para cada una de las etapas, identificando cual es la contribución científica que presenta esta tesis, su implementación en la práctica y el trabajo futuro a desarrollar.

1.5. Alcances

El producto final de este trabajo se enmarca en la detección de tendencias en el nicho de la tecnología a nivel mundial. Para ello, se espera haber desarrollado un modelo teórico de detección de tendencias a través de análisis de noticias y opiniones en la Web, y un software capaz de llevar el modelo a la práctica. Si bien las posibilidades de investigación para este tipo de modelos son grandes, este trabajo tiene principalmente como objetivo el desarrollo de un prototipo que muestre el potencial de esta rama de investigación y que valide el concepto propuesto. Además, se espera

validar el modelo a través de experimentos analizados con métricas conocidas y opiniones de expertos en tendencias tecnológicas. Finalmente, se considera que la herramienta debe soportar la inclusión *a posteriori* de otros nichos (por ej. *Minería, Educación, etc.*) y otras fuentes de opiniones además de *Twitter*.

1.6. Resultados esperados

Al termino de este trabajo, se espera que el modelo de detección de tendencias, el cual estará compuesto tanto del modelo de opiniones como de el modelo de tópicos, sea capaz de dar indicios de las tendencias venideras en el mundo de la tecnología a nivel mundial. Si bien el modelo no determinará de manera concreta cuales son los productos y servicios que marcarán tendencia a futuro, este sí será capaz de mostrar qué tópicos están en boga, cuáles van aumentando su presencia e impacto en la Web, y los contenidos en los medios que están asociados a estos, de manera que el grupo de expertos que utilice esta herramienta de apoyo sea capaz de juzgar con muchos mejores argumentos qué acciones tomar a nivel operacional y estratégico para hacer frente a las acciones de la competencia.

1.7. Contexto de la Investigación

Este trabajo de investigación es financiado por la consultora Duam - Innovación al Sur del Mundo, ubicada en Santiago de Chile, con el fin de desarrollar una herramienta que apoye el desarrollo de sus proyectos de inteligencia de mercado.

Dentro de las estrategias de Duam para captar nuevos clientes, posee un equipo especializado en el desarrollo de proyectos de inteligencia de mercado, disciplina emergente que tiene como objetivo principal el detectar posibles amenazas, nuevos entrantes y oportunidades de inversión o expansión.

Para llevar a cabo sus estudios de inteligencia de mercado, los miembros del equipo realizan periódicamente un análisis exhaustivo del mercado en el cual se encuentran inmersas las empresas o los productos que se deseen estudiar. Este análisis actualmente se realiza de forma manual, proceso que se vuelve infactible de realizar si se desea trabajar en varios proyectos a la vez, o si un proyecto en particular abarca muchas fuentes de información, razón por la cual nace la necesidad de automatizar parte del proceso, sobre todo la recopilación de información desde las distintas fuentes que se estén analizando.

1.8. Estructura de la tesis

En el capítulo que se presenta a continuación, se definirá el marco conceptual sobre el cual se trabajará a lo largo de esta investigación, y además, realizará una revisión bibliográfica extensa, donde se hablará del estado del arte en técnicas y algoritmos sobre extracción de tópicos, web opinion mining e identificación de tendencias en la web.

En el capítulo 3, se presentan los algoritmos de extracción de tópicos que serán utilizados en esta investigación, señalando como serán adaptados para la detección de tendencias en la web.

A lo largo del cuarto capítulo, se describe el modelo propuesto para extraer opiniones de documentos de *microblogging* presentes en un flujo continuo de información.

La contribución principal de este trabajo de investigación se presentan en el capítulo 5, donde se detalla el modelo de detección de tendencias, y los distintos algoritmos involucrados en este, junto con posibles aplicaciones, escenarios de práctica, y sugerencias de configuración para su uso.

En base a la investigación realizada, en el capítulo 6 se describe la configuración experimental en los que se llevará acabo la aplicación práctica del modelo propuesto. A su vez, se describen métricas de evaluación tanto para la extracción de opiniones como para la detección de tendencias.

A partir de los experimentos definidos en el capítulo anterior, en el capítulo 7 se muestran los principales resultados para el modelo de detección de tendencias que se ha propuesto. Estos resultados se muestran de acuerdo a las métricas y a los criterios de evaluación ya definidos.

Finalmente, en el capítulo 8, se dan a conocer las conclusiones principales de esta investigación, entre las cuales se destacan las contribuciones más relevante y una propuesta para trabajo futuro y posibles líneas de investigación a considerar.

1.9. Contribuciones

A partir de la investigación realizada en este trabajo se han desarrollado las siguientes publicaciones:

- **Revista de Ingeniería de Sistemas:** El paper titulado *Una aplicación de Web Opinion Mining para la extracción de tendencias y tópicos de relevancia a partir de las opiniones consignadas en blogs y sitios de noticias* ha sido aceptado y está en proceso de publicación.
- **Workshop On Social Web Intelligence:** Se ha publicado un paper en este workshop titulado *Sentiment Polarity of Trends on the Web Using Opinion Mining and Topic Modeling.*
- **Detecting Trends on the Web: A Multidisciplinary Approach**: Está en proceso de revisión para ser enviado al journal Knowledge and Information Systems

Capítulo 2

Marco Conceptual

En este capítulo se dará a conocer la revisión bibliográfica realizada para definir el marco conceptual necesario para entender, tanto el problema presentado en este trabajo de investigación, como la solución que se describirá en detalle en los siguientes capítulos. Además, se realizará una descripción detallada de cómo se han tratado de abordar en la literatura académica cada uno de los problemas a desarrollar en esta tesis. Para lograr este objetivo de la mejor manera posible, se partirá describiendo de lo general a lo particular en cada una de las secciones que se mencionan a continuación.

Debido a la naturaleza del problema descrito en el capítulo anterior, y por razones que se darán a conocer en el capítulo 3, se considera que las ramas de investigación relevantes para este trabajo son las de *recuperación de la información*, *minado de opiniones* y *detección de tópicos*. Además, es de especial importancia tratar bibliografía que hable sobre el procesamiento de documentos generados en plataformas de *microblogging*, los cuales si bien de acuerdo a lo expuesto por Aditya en [32] conllevan una serie de problemas por sus limitaciones de tamaño, son una de las mayores fuentes públicas de documentos opinados en la actualidad.

En esta búsqueda de conocimiento en la Web y en particular para poder detectar tendencias a través de información presente en ella, es necesario conocer qué se está discutiendo en todos los sitios web que se consideren relevantes para el nicho en observación. En general, se considera importante observar la blogosfera, los sitios de noticias y las redes sociales, ya que son estas tres fuentes de información las principales a la hora de determinar qué temas son tratados en el día a día y cuáles de estos son *hot-topics*. Con esto en mente, para poder extraer conocimiento a partir de documentos presentes en la web, es necesario dar a conocer el área de *recuperación de la información*, con especial enfoque en la recuperación de documentos desde sitios de noticias y blogs, junto con documentos de *microblogging* debido a su gran valor para los algoritmos de minado de opiniones.

Una vez que se han recuperado los documentos necesarios para realizar la extracción de conocimiento a partir de ellos, es necesario saber qué es lo que se está comentando en los sitios de noticias y

en los blogs. Debido a la gran cantidad de documentos que se espera recuperar, es necesario agruparlos para enfocar el análisis en lo relevante, para lo cual se pretende utilizar un modelo de extracción de tópicos con el fin de realizar esta agrupación por tópicos en discusión en la web.

Finalmente, para cerrar el ciclo que busca obtener indicios sobre la demanda futura sobre productos o servicios a través de la información presente en la web, es necesario conocer qué es lo que opinan sus usuarios sobre los tópicos previamente extraídos, lo que genera la necesidad de dar a conocer la rama del recuperación de la información llamada *web opinion mining*, la cual se enfoca en tratar de extraer el conocimiento vertido en la web por los usuarios haciendo uso de las opiniones vertidas por estos en las redes sociales.

2.1. La Web y fuentes de documentos

Tal como se menciona en el capítulo uno, el concepto tras la Web fue propuesta originalmente por Tim Berners-Lee en el año 1989 [6], con el propósito de dar una plataforma robusta para facilitar el acceso distribuido de documentos asociados a los distintos experimentos que se realizaban en el CERN en ese tiempo. La propuesta inicial, consideraba la implementación de un sistemas de información enlazada, con *nodos* que simbolizaban datos o entidades e *hipervínculos* que indicaban la relación entre estos.

A pesar de esto, sin la aparición de la *Internet*, la visión de Tim Berners-Lee nunca hubiese llegado a ser lo que es hoy en día la *World Wide Web*. La *Internet* es una red de computadores a nivel global, los cuales hacen uso del protocolo *TCP/IP* para comunicar datos entre sí. Haciendo una comparación con lo propuesto en el CERN el año 1989 y la World Wide Web, la *Internet* es la arquitectura que da soporte al sistema de información enlazada, cada integrante de la Web es un nodo de datos o una entidad, y las relaciones entre estos se dan a conocer a través de hipervínculos.

Esta interconexión entre sistemas es lo que da paso a una serie de cambios importantes en la manera en que la información se comparte y la gente interactúa, disminuyendo de manera significativa los costos y los tiempos en casi todos los tipos de comunicación que se usan hoy en día, y aumentando considerablemente la cantidad de información disponible tanto para las empresas como para el usuario común.

2.1.1. La Web 2.0

Con el paso del tiempo, debido a la evolución de la tecnología presente tanto en el lado de los servidores como en el del usuario, la Web fue adquiriendo roles y capacidades que inicialmente no estaban consideradas, tales como el empoderamiento de este último en la creación de contenido y el

diseño de una gran cantidad de sitios web centrados en el usuario, tales como las redes sociales y los *wiki*. Este nuevo enfoque de la Web, llevó al nacimiento del término Web 2.0, que hace referencia al cambio de paradigma desde una Web de sólo lectura a una Web donde es posible, para los usuarios, tanto publicar como recibir contenido.

En la sociedad de hoy, donde la existencia de la Web 2.0 ha tomado un rol protagónico en la manera en que las personas se comunican e interactúan entre si, han nacido múltiples herramientas y sitios web que facilitan este proceso. Entre los sitios que facilitan la creación de contenido por parte de los usuarios, se encuentran los sitios de *blogging* y *microblogging*, los cuales tienen como objetivo principal que las personas o comunidades compartan información con sus pares o de manera pública en la web; y entre los sitios que permiten la comunicación entre los usuarios destacan las llamadas *redes sociales*, tales como *Facebook*[1] y *Google+*[2] entre otros.

Cabe destacar, que los documentos que se generan actualmente en los sitios web, pueden ser de carácter objetivo (e.j. un artículo informando sobre un hecho en particular o un enlace a contenido externo) o subjetivo (e.j. una opinión sobre un artículo, una conversación entre amigos). Es en esta dualidad en donde yace una gran oportunidad para los sistemas de extracción de conocimiento, permitiendo complementar la información objetiva con las opiniones de los usuarios de estos sitios.

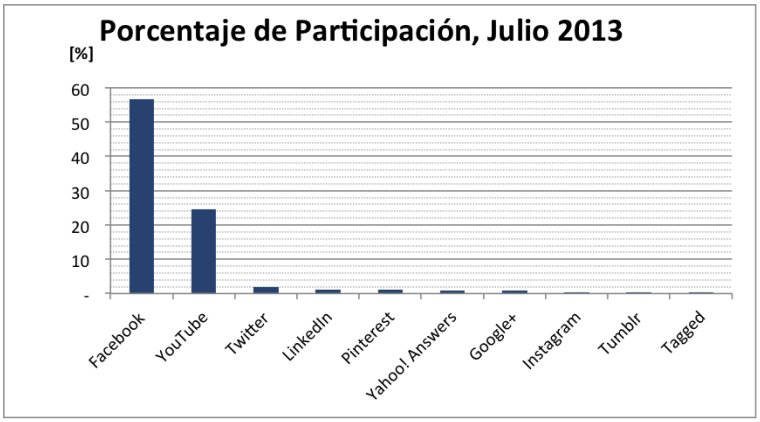

Figura 2.1: Participación de mercado de los principales actores de la Web 2.0.

En la figura 2.1[3] es posible observar la evolución del panorama competitivo en la Web 2.0 durante el último tiempo, donde se puede distinguir de manera clara el aumento sostenido de participación de los sitios considerados como redes sociales, tales como Facebook y YouTube, y las plataformas de microblogging como Twitter, y además, la importancia que estos han tomado en los patrones de uso de la Web.

A continuación, se da una pequeña introducción a cada tipo de sitio web relevante que se puede

[1] `http://www.facebook.com`
[2] `http://plus.google.com`
[3] Fuente: Experian Hitwise

encontrar hoy en día en la Web 2.0. En estas, daremos a conocer su génesis, estado actual y los sitios más emblemáticos de cada uno de ellos.

2.1.2. Blogging

Para definir que se considera como microblogging, es necesario partir por su símil el blogging. Este es definido como el acto de publicar un documento, comúnmente conocido como artículo o entrada en un blog. De acuerdo a lo presentado por Rebecca Blood [11], un *blog* es una bitácora virtual donde uno o más individuos publican documentos de variados tipos para que estén disponibles de manera pública en la web. En general, los documentos de un blog son presentados en orden cronológico de acuerdo al momento en que fueron publicados.

En los blogs existentes en la Web, cada documento está compuesto de texto y en variadas ocasiones cuentan con respaldo multimedial (fotografías, videos, audio, etc.). Dependiendo de la entidad que publica documentos en un blog, la información presente en estos puede ser opinada, como es en el caso de blogs de opinión o de críticas; o no opinada, como es el caso de los blogs de noticias.

A la hora de explorar el panorama competitivo en el mundo del blogging desde el punto de vista de la recuperación de la información, es importante analizar la potencialidad de los blogs como fuente importante de todo tipo de documentos, para lo cual es necesario mencionar cuales son los actores importantes en el mundo del blogging, es decir, qué blogs son los más leídos en el mundo.

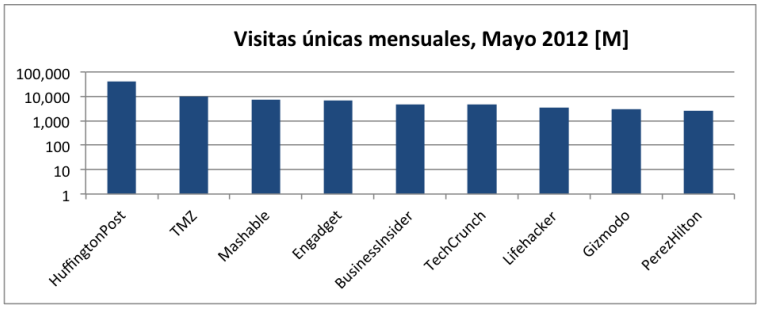

Figura 2.2: Participación de mercado de los principales sitios de blogging.

En la figura 2.2[4] se puede apreciar que las plataformas de **blogging** más utilizadas durante el mes de Mayo del 2012:

- **Huffington Post** con **41 millones** de visitantes únicos.
- **TMZ** con **9.9 millones** de visitantes únicos.
- **Mashable** con **7.5 millones** de visitantes únicos.

[4]Fuente: Elaboración propia con información recoletada de múltiples fuentes.

- **Engadget** con **6.8 millones** de visitantes únicos.
- **Business Insider** con **4.6 millones** de visitantes únicos.

Si bien, tal como se desprende de la información que se presentará en referencia al uso de las redes sociales y de los sitios de microblogging, un blog por si sólo no es tan relevante debido a la cantidad de usuarios que este posee, en su conjunto sí son una fuente a considerar de documentos debido a la gran variedad de temas que son cubiertos por ellos, y inmensa diversidad de opiniones que se pueden encontrar en sus artículos. Además, otra variable importante a considerar a la hora de cuantificar qué significan estas cantidades de visitas presentadas en la figura 2.2, hay que tener en cuenta que gracias a la existencia de las fuentes sindicables estas son sólo una parte de la cantidad real de usuarios que ingresan a un blog en particular.

2.1.3. Microblogging

Por otro lado, el *microblogging*, según [34], si bien busca suplir herramientas para una necesidad similar a aquella que los blogs satisfacen, se diferencia en el hecho de que los *microblogs* se enfocan en compartir *pequeños elementos* de contenido tales como frases cortas, enlaces a sitios web, videos, imágenes y otros.

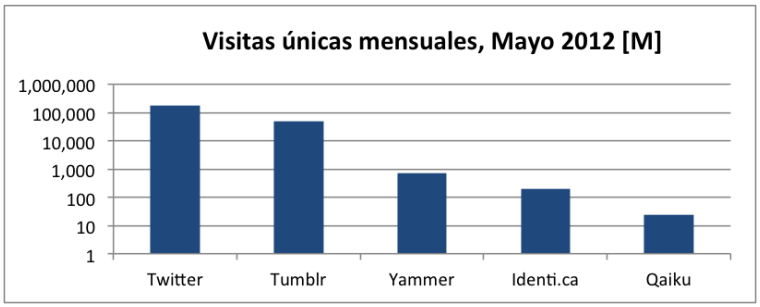

Figura 2.3: Participación de mercado de las principales sitios de microblogging.

En la figura 2.3[5] se puede apreciar que las plataformas de **microblogging** más utilizadas durante el mes de Mayo del 2012 son:

1. **Twitter** con **170 millones** de visitantes únicos.
2. **Tumblr** con **50 millones** de visitantes únicos.
3. **Yammer** con **700 mil** vsitantes únicos.
4. **Identi.ca** con **200 mil** visitantes únicos.
5. **Qaiku** con **24 mil** visitantes únicos.

[5]Fuente: Elaboración propia con información recoletada de múltiples fuentes.

2.1.4. Redes Sociales

De acuerdo a Ellison y Boid [13], un sitio web se define como una red social cuando "permiten que un individuo construya un perfil público o semi-público dentro del sistema; construir una lista de usuarios con los que desea compartir una conexión o información; y, ver y recorrer la lista de conexiones que él y sus contactos hayan realizado.". Una característica importante de las redes sociales es que tienen la capacidad de generar contenido opinado, es decir, que cada usuario tiene la habilidad de dar a conocer su opinión sobre un hecho o un objeto en particular y que los demás integrantes de la red social comenten sobre ella y den su aprobación o rechazo.

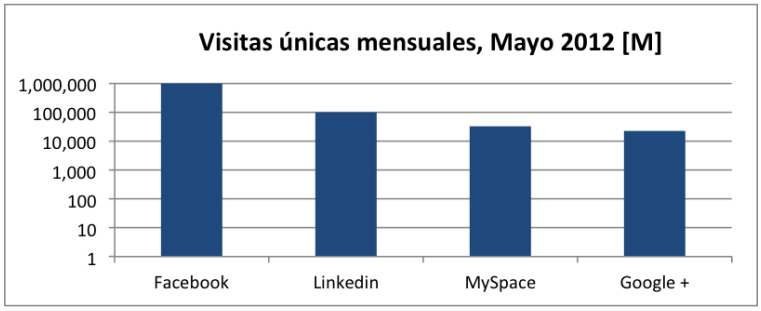

Figura 2.4: Participación de mercado de las principales redes sociales.

En la figura 2.4[6] se puede ver que las **redes sociales** más utilizadas durante el mes de Mayo del 2012 son:

1. **Facebook** con **980 millones** de visitantes únicos.
2. **LinkedIn** con **97 millones** de visitantes únicos.
3. **MySpace** con **34 millones** de visitantes únicos.
4. **Google+** con **22.3 millones** de visitantes únicos.

De estos datos, es importante destacar que si bien Google+ tiene pocos visitantes únicos, sigue siendo un actor importante debido a la gran cantidad de servicios que Google tiene integrados junto a su red social, y por otro lado, si bien MySpace tiene una gran cantidad de usuarios, su popularidad ha ido decreciendo en el último año.

2.2. Recuperación de la información

De acuerdo a lo presentado por Baeza *et al.* en [4], la recuperación de la información es la ciencia que se encarga de "representar, almacenar, organizar y dar acceso a información". Además, se busca

[6]Fuente: Elaboración propia con información recoletada de múltiples fuentes.

que la manera de representar y organizar la información sea sencilla y eficiente para el usuario. En particular, para este trabajo se buscará recuperar información a partir de artículos de noticias y documentos presentes en redes de *microblogging*, los cuales al contener lenguaje natural, presentan el desafío de no ser representables de manera estructurada [47].

Para efectos de este trabajo de investigación, se propone la siguiente definición de *documento*:

Definición 2.1. Un *documento d* es una unidad de información que puede ser estudiada y que además es considerada ya sea como evidencia de un hecho o como la expresión de una percepción subjetiva de uno por parte de una entidad dentro de un sistema. Ejemplos de documentos son: un artículo en una revista, un paper científico, una columna de opinión escrita por un periodista en un diario, etc.

2.2.1. Recuperación de documentos desde la web

El área de la recuperación de la información no es posible de concebir sin diseñar previamente una manera de recorrer la web buscando documentos que además almacene estos de manera local para su posterior análisis. Es gracias a esta necesidad primordial de los sistemas de recuperación de la información, que múltiples soluciones para recuperar documentos han tomado forma en los últimos años, las cuales todas se basan en el concepto de *Web Crawling* que se describirá a continuación.

Web Crawling

Un *web crawler* o siplemente *crawler* es un software que actúa como agente visitante dedicado a recorrer la web para recuperar todos los documentos que sean necesarios a la hora de realizar un análisis de la web, este recorrido lo realiza aprovechando la estructura de grafo que posee la web, lo que le permite partiendo de uno o más puntos alcanzar una gran cantidad de nodos en ella sin la necesidad de ser operado de manera supervisada.

Son de mayor importancia en los procesos de recuperación de la información, ya que aquellos algoritmos parten de la premisa que poseen el documento a su disposición, y por lo tanto, si se desea recuperar información desde la Web, es necesario poseer una copia local de estos. Debido a esto los *crawlers web crawlers* son utilizados en la mayoría de las plataformas de recuperación de la información [29].

Además, los *crawlers* son usados en una gran gama de plataformas y servicios que se basan en poseer grande cantidades de documentos para extraer información desde estos, entre las aplicaciones más importantes de los crawlers podemos encontrar:

- Motores de búsqueda
- Portales enfocados en un tópico en particular

- Seguimiento de marca en la web

Dependiendo de la naturaleza del crawler, la cual se ve descrita por lo que este busque recuperar desde la web, el nivel de detalle que se quiere lograr, la manera en que recorra la web y cualquier otra restricción que influya en cómo y qué recupera de la web, estos pueden ser clasificados en alguna de las siguientes categorías:

- **De propósito general**:

  Son aquellos utilizados por motores de búsqueda en la Web[7], su particularidad consiste en que utilizan un conjunto de fuentes como punto de partida para la recuperación de documentos, a partir de la cual realizan una búsqueda en profundidad para visitar la mayor cantidad posible de nodos en el grafo sobre el cuál trabajan.

- **Focalizados o temáticos**:

  Estos son una versión especializada de los crawlers de propósito general que tienen como objetivo minar documentos que solamente pertenencen a una temática en particular o a un subconjunto acotado de sitios.

- **Distribuidos**:

  Son todos aquellos crawlers que se encuentren distribuidos en una red de computadores para aumentar su capacidad de procesamiento. Uno de los principales desafíos de este tipo de crawler es sincronizar todos los nodos en acción para evitar información incorrecta y optimizar el flujo de datos del sistema en su totalidad.

En la figura 2.5 se puede apreciar un flujo básico descrito por Pant *et al.* [57] el año 2004, que es comúnmente utilizado en un el desarrollo de crawler secuencial. A continuación se detalla cada uno de los pasos importantes en el ciclo de vida de un crawler:

1. **Inicialización de la frontera con URLs semillas**: Para comenzar, es necesario dar a conocer las definiciones de *feed*, *URL* y *frontera*:

   Definición 2.2. Un *feed* F se define como una fuente de documentos d ordenados de manera cronológica y amparados bajo una misma temática.

   Definición 2.3. Una *url* U es un identificador único de un recurso en la web que permite acceder a este a través de los distintos protocolos presentes en la Internet. En particular, un feed es descrito por una URL.

[7] Ejemplos de motores de búsqueda son Google Search, Bing y DuckDuckGo entre otros.

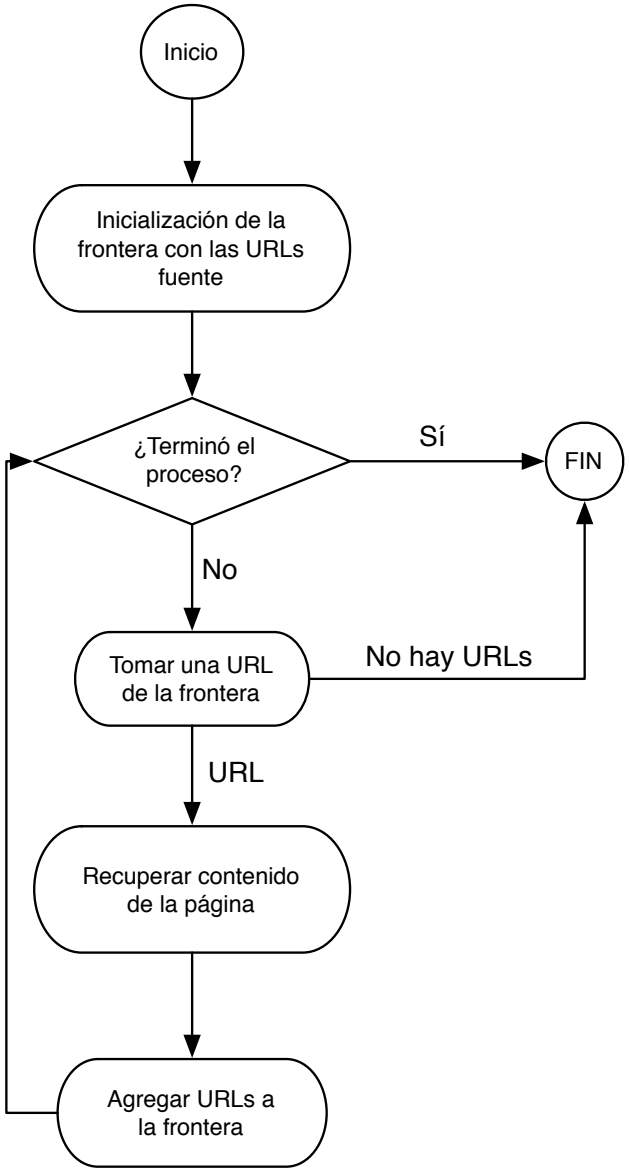

Figura 2.5: Flujo de un crawler secuencial

Definición 2.4. La ***frontera*** es un conjunto $\{U_i\}_{i\in\mathbb{N}}$ de URLS, la cual señala qué recursos de la web debe visitar el crawler a lo largo de su ejecución.

Se le llama inicialización de la frontera al proceso de cargar esta para su posterior lectura y procesamiento. Dependiendo de la cantidad información contenida en la frontera, y para favorecer la velocidad de recuperación de documentos, es posible cargar las entradas de la frontera en memoria, o utilizar una estrategia compartida de cargar parcialmente esta e ir rotando las fuentes presentes en memoria en caso de que existan recursos limitados.

2. **Recuperación de documentos**: En esta etapa, se realiza la recuperación de documentos de cada URL presente en la frontera y se almacenan para su posterior procesamiento. A

continuación se presentan dos algoritmos de recuperación de documentos que son ampliamente utilizados en crawlers.

- **Naive Best-First**: Para su implementación, la frontera se describe como una cola de prioridad basada en un score de similitud acorde a una representación vectorial de las frecuencia de palabras presentes en los documentos. En la primera iteración el algoritmo realiza una comparación a través de la métrica similitud coseno entre el documento y la descripción dada por el usuario y asigna como *score* a cada una de las URLs presentes en el documento la similitud entre ambos.

  Definición 2.5. La similitud coseno entre dos vectores A y B se calcula como sigue:

$$sim(A, B) = \frac{A \cdot B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}}$$

  En las próximas iteraciones el crawler obtiene las URLs a visitar a partir de la cola y procede a calcular el *score* de similitud entre el la página padre (la que corresponde a la URL) y las URLs sin visitar que se extrajeron del documento. En particular, la similitud entre una página p y una query q, considerando v_p como y v_q como el vector de frecuencias mencionado anteriormente se calcula como $sim(v_p, v_q)$. Para más detalles ver el algoritmo 2.2.1

3. **Parseo de los documentos recuperados**: En caso de que los documentos recuperados no sean texto plano, se procede a realizar un parseo de estos extrayendo todo el texto y la metadata que se requiera. Por ejemplo, en el caso de crawlers de propósito general, se extraen todos los enlaces que se encuentre en el documento; si se utiliza un crawler focalizado y el documento proviene de un blog, se procede a extraer el autor, los tags, la categoría a la que pertenece, etc.

4. **Procesamiento de documentos**: El procesamiento de documentos en el área de recuperación de la información será descrito en detalle en la sección 2.2.3.

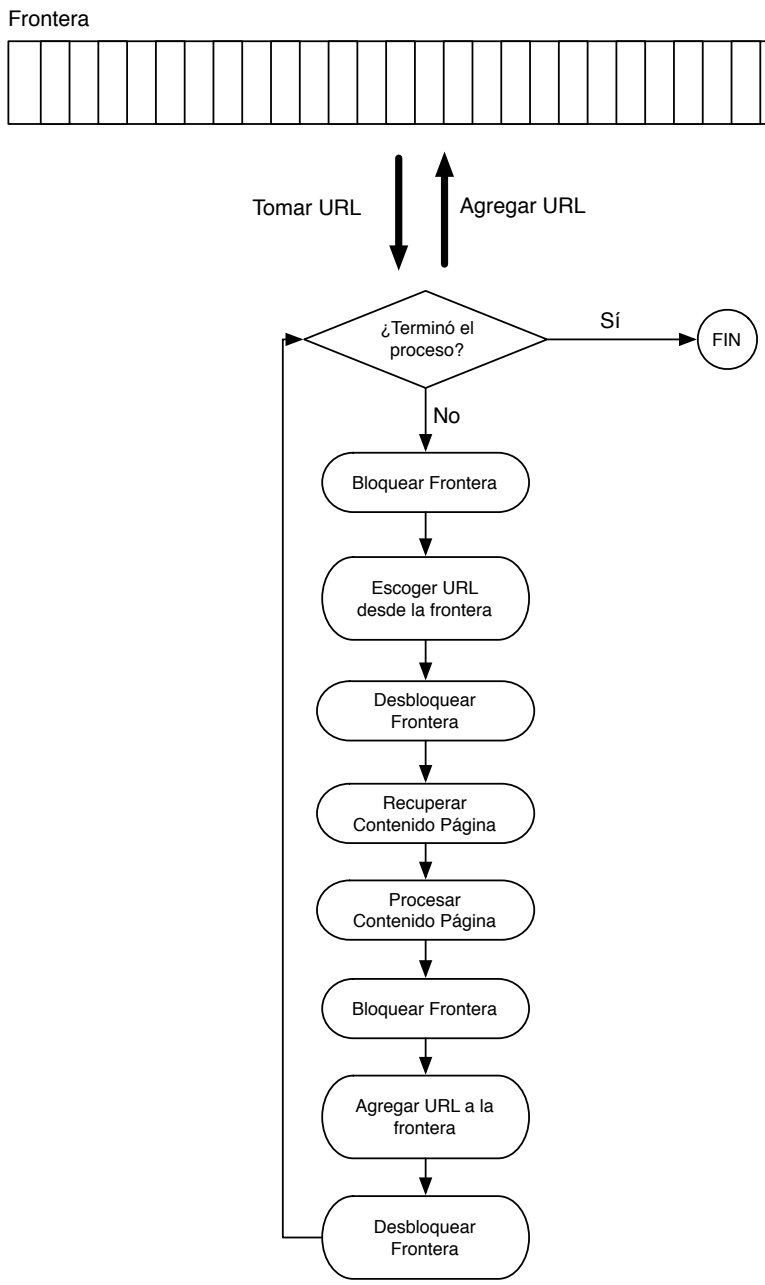

Figura 2.6: Flujo de un crawler utilizando el algoritmo *Naive Best-First*

5. **Alimentar la frontera con nuevas URLs semillas**: Dependiendo del tipo de crawler que se esté implementando, es posible que luego de realizar todas las tareas previas sobre los documentos, sea necesario alimentar la frontera con nuevas URLs, y repetir el ciclo nuevamente, para así lograr una mayor cobertura del grafo de la web. Este paso es uno de los más importantes en los crawlers utilizados por los motores de búsqueda, ya que es el que permite que a partir de sólo unos miles de sitios semilla se indexe gran parte de la web

Algoritmo 2.2.1: Naive Best-First Crawling

Data: $\{U_i\}_{i \in \mathbb{N}}$

Inicializar la frontera $Fr = EmptyPriorityQueue$;
for $i \leftarrow 1$ **to** $\|\{U_i\}_{i \in \mathbb{N}}\|$ **do**
    Descargar documento presente en U_i y almacenarlo en la variable d_i;
    $Fr.add(U_i, sim(U_i^d, d_i))$ con U_i^d la descripción de la URL i;
    for *hipervínculo* $h \in d_i$ **do**
        $Fr.add(h, sim(\vec{d_i}, \vec{h}))$;
while $len(Fr) \neq 0$ **do**
    $URL = Fr.pop()$;
    $d = Descargar(URL)$;
    for *hipervínculo* $h \in d$ **do**
        $Fr.add(h, sim(\vec{d}, \vec{h}))$;

Es importante señalar que no toda la web puede ser indexada a través de los métodos más utilizados, ya que existen muchas web aisladas y que por consiguiente no son alcanzables a través de los arcos disponibles en la web. Existen, a grandes rasgos, dos subconjuntos de la web que no son alcanzables por crawlers convencionales, el primero, es la llamada *Deep Web* o *Dark Web*, la cual está compuesta por todos aquellos sitios que no son alcanzables a través de hipervínculos de texto, páginas generadas dinámicamente, sitios referenciados atravéś de contenido multimedia, sitios privados y sitios sin enlaces caen dentro de esta categoría; y por otro lado se encuentra la *Dark Internet*, que hace referencia a todos los *hosts* inalcanzables en *Internet*.

Además del enfoque basado en URLs, para algunas aplicaciones es posible utilizar un enfoque distinto, que no se basa en aprovechar la estructura de grafo de la Web y por consiguiente, ignora los *hipervínculos* presentes en los documentos recuperados, si no que se basa en utilizar la frontera como una lista de sitios a recorrer en búsqueda de documentos, lo que es comúnmente llevado a la práctica a través del uso de fuentes de documentos por suscripción, las cuales serán descritas a continuación.

2.2.2. Fuentes de documentos por suscripción

Las fuentes de documentos por suscripción fueron tomando forma a medida que la información presente en la web fue aumentando de manera considerable, se han desarrollado múltiples tecnologías que facilitan el acceso por parte de los usuarios de la Web a la información presente en sus sitios,

entre las más utilizadas en la actualidad se encuentran:

RSS y Atom

RSS[8] y Atom son formatos utilizados en la creación de feeds web, las cuales se enfocan en publicar documentos que siguen una secuencia temporal a través de un formato estándar. En la actualidad ambos formatos son utilizados frecuentemente por blogs, noticiarios y podcasts[9] entre otros, lo que ha permitido que los usuarios tengan acceso a una mayor cantidad de contenido sin invertir una gran cantidad de tiempo en ello. Además, la capacidad de enviar contenido al usuario de manera automática favorece a los sitios que tengan acceso a sus documentos en este formato, ya que aumenta el nivel de fidelidad de los usuarios y reduce el nivel de fuga.

Toda fuente que tenga a disposición sus documentos en uno de estos formatos se considera una fuente sindicable. En otras palabras:

Definición 2.6. Se dice que un fuente F es sindicable si se dispone de un punto de entrada único en donde se expongan documentos pertenecientes a esta en orden cronólogico, ya sean estos asociados a una ventana de tiempo específica o no. En particular, toda *fuente* disponible *RSS* o *Atom* se considera sindicable.

Una característica importante de la mayoría de las fuentes sindicables, es que los documentos expuestos en el punto de entrada son temporales, lo que significa que el conjunto de documentos $\{d_i^F\}_{i\in\mathbb{N}}$ presente al recuperar la lista de documentos desde la fuente F depende directamente del instante t en el cual se realice esta revisión, por lo tanto, es posible definir el conjunto $\{d_i^{F^t}\}_{i\in\mathbb{N}}$ como la lista de documentos disponibles en una fuente F en un instante t.

Newsletters

Las *newsletters* son un modelo de sindicación de noticias a través del cual las noticias son enviadas de manera periódica al correo electrónico de los usuarios que se han suscrito a estas. En el comienzo de la era de los equipos blackberry y smartphones tuvieron su mayor popularidad, debido a la gran utilidad de poder leer noticias mientras se está viajando o no se tiene un computador cerca, sin embargo, con el paso del tiempo y el aumento de la cantidad y calidad de las aplicaciones disponibles para los distintos sistemas operativos de smartphones, las feeds sindicables en formato RSS o Atom se han convertido en la fuente primordial de noticias.

2.2.3. Procesamiento de documentos

Dependiendo del contexto en el que se vayan a utilizar los documentos recuperados por un crawler, es posible que se requiera realizar un procesamiento de estos en orden de lograr que el

[8] RSS es un acrónimo de Really Simple Syndication

[9] Un archivo multimedia destinado a ser distribuido a través de la web es llamado podcast

conjunto de documentos sea analizable por los algoritmos que se aplicarán sobre ellos. En general, todos los algoritmos de procesamiento de datos apuntan a realizar una reducción dimensional de la data recolectada, apuntando a disminuir la dispersión estadística de los datos.

Lematización o Stemming

El proceso *stemming* consiste en la remoción de los sufijos en palabras que pertenezcan a la misma familia semántica. El objetivo tras remover sufijos de palabras con significados similares es reducir la cantidad de dispersión de los conceptos presentes en un documento, y por consiguiente, aumentar la precisión de los algoritmos de recuperación de la información que se utilicen para extraer conocimiento del documento en cuestión.

El algoritmo de *stemming* más utilizado en la actualidad, es el algoritmo de Porter [60], el cual considera algunas definiciones para desarrollar el algoritmo de lematización propuesto.

Definición 2.7. En el idioma Español una *consonante* dentro de una palabra es cualquier letra del alfabeto que no sea A, E, I, O y U. Por otro lado, en el inglés una consonante es cualquier letra que no sea A, E, I, O, U, ni Y precedida por una consonante. Además, una *vocal* es cualquier letra que no sea considerada una consonante.

Definición 2.8. La *medida* de una palabra, está descrita por la cantidad de consonantes precedidas por vocales dentro de si, es decir, si denotamos **V** a las vocales y **C** las consonantes, la medida m de una palabra viene dada por:

$$[C](VC)^m[V]$$

Un ejemplo de medida es el siguiente: para la palabra *pulula*, tiene medida $m = 2$ ya que esta puede ser decompuesta como P - ULUL - A.

Luego, para determinar que sufijos deben removerse, se define una serie de reglas dependientes del idioma en el cual se encuentra el documento, las cuales tienen todas la siguiente forma:

$$(condicin)S1 \rightarrow S2$$

Lo que significa que si la palabra termina con el sufijo $S1$, y la raíz previa a $S1$ satisface la condición dada, entonces $S1$ es reemplazado por $S2$. En general, la condición para el reemplazo de sufijo se define en base a la *medida* de la palabra, y puede estar compuesta de operadores lógicos *y* y *o* además de las siguientes condiciones presentadas en el paper original de Porter, las cuales sólo aplican al inglés:

- ***S**: La raíz termina con S o cualquier letra que se requiera.
- ***v***: La raíz contiene una vocal.
- ***d**: La raíz termina con una doble consonante

- ***o**: La raíz termina en CVC (consonante, vocal, consonante), donde la segunda consonante no es W, X o Y.

Si bien el algoritmo de Porter se enfoca sólo al inglés, es posible derivar un conjunto de reglas similares que caractericen a cualquier lenguaje, y por consiguiente, extender su algoritmo para realizar lematización en español.

Remoción de stop-words

Además de la lematización, otro proceso de suma importancia en la recuperación de la información es la remoción de *stop-words*, proceso que también ayuda a reducir el sesgo estadístico. Para explicar la utilidad de este proceso, y la forma en que se lleva a cabo, primero es necesario definir lo que se considera como una *stop-word*.

Definición 2.9. Stop-word: Una *stop-word* es una palabra w tal que su frecuencia de uso en el lenguaje al que pertenezca este muy por sobre la media, o cuyo significado sea neutro en relación a lo que se está estudiando. Un clásico ejemplo de stop-words son los artículos: *el, la, los, las, etc.*

Considerando esta definición de stop-word, es posible dar base a la realización de este proceso en muchas aplicaciones de recuperación de la información: su uso disminuye el ruido en la información. A continuación se presenta un algoritmo clásico que se encarga de remover stop-words de un documento.

Sea $\{sw_i\}_{i\in\mathbb{N}}$ el conjunto de stop-words a utilizar, _ el carácter espacio, ϵ el string vacío, $\cdot$ el operador utilizado para concatenar strings y d un documento del cual se desean remover las stop-words. Entonces, el algoritmo a ocupar se describe como sigue:

Algoritmo 2.2.2: Remoción de stop-words

Data: $\{sw_i\}_{i\in\mathbb{N}}, d$

Result: d'

```
1 d' = ϵ;
2 for w ∈ d do
3     if w ∉ {sw_i}_{i∈ℕ} then
4         d' = d'·␣·w
```

Hay otras variantes basadas en expresiones regulares y otras funcionalidades facilitadas por los distintos lenguajes de programación utilizados para llevar a la práctica el algoritmo, pero el algoritmo presentado es la base de todos estos, y por consiguiente, es posible como un ejemplo válido para la mayoría de los algoritmos existentes.

2.3. Modelos de Tópicos

De acuerdo a Blei *et al.* [10], los modelos de tópicos buscan dar solución al problema de modelar colecciones de documentos y cualquier otro tipo de datos discretos. Su principal objetivo es reducir la cantidad de información necesaria para dar una descripción acabada de ellas, y permitir así el procesamiento más eficiente de estas, sin sacrificar las relaciones estadísticas inherentes de cada colección. Por lo tanto, un tópico es aquel conjunto de términos capaces de representar, sin pérdida de información estadística, un tema en particular tratado en una colección de documentos.

Un modelo de tópicos es una construcción estadística capaz de modelar las relaciones subyacentes entre las palabras, los documentos y la colección como un todo, con el fin de descubrir los temas principales que se encuentran en una colección de documentos, como estos se relacionan y cómo van cambiando a lo largo del tiempo. Su principal uso es el organizar colecciones de documentos que con otros algoritmos de clustering no tendrían suficiente cohesión como para obtener clusters que sean representativos de la información contenida en esta colección. Además, debido a la gran versatilidad que otorgan estos modelos, en los últimos años han sido adaptados para organizar colecciones de muchos tipos de información como por ejemplo, información genética, imágenes y audio entre otros.

A la hora de analizar un modelo de tópicos desde un punto de vista crítico, es necesario conocer ciertas definiciones básicas que son transversales a todos ellos. Primero daremos definiciones desde un punto de vista lingüístico y luego desde un punto de vista teórico a través de los distintos modelos que se describirán en esta sección.

Definición 2.10. Un *tópico* t se define como el sujeto que es caracterizado o tratado en un texto, discurso o conversación, es decir, un objeto sujeto a discusión.

Definición 2.11. Un documento d trata sobre un tópico t si da a conocer información sobre este, o en otras palabras, si d contiene conceptos relacionados semánticamente con t. Un documento puede tratar sobre más de un tópico y un tópico puede ser tratado en más de un documento.

A lo largo de los últimos años, una gran variedad de modelos han sido desarrollados con el objetivo de solucionar este problema, de los cuales los más utilizados son los creados por Blei *et al.*. Por un lado, se encuentran el modelo estático *Latent Dirichlet Allocation* (LDA) [10], el cual considera que las palabras son intercambiables entre documentos y que no hay correlación entre cada tópico; el modelo estático *Correlated Topic Model* [8], el cual incluye el hecho de que los tópicos se correlacionan entre sí y que cada palabra pertenece a un documento en particular; y finalmente el modelo dinámico *Dynamic Topic Model* [9], que busca modelar la relación entre documentos y tópicos en una colección de documentos con una componente temporal, permitiendo analizar la evolución de un tópico a lo largo del tiempo y la manera en que cada documento colabora con esta.

2.3.1. Latent Dirichlet Allocation

Para comenzar, se discutirá el modelo llamado *Latent Dirichlet Allocation* [10] el cual es considerado como el más sencillo de los modelos de tópicos presentes hoy en día, y por ello permite obtener un primer acercamiento a estos para introducir definiciones importantes y lograr obtener nociones relevantes a la hora de comprender modelos más avanzados.

A continuación, se definen los conceptos bases necesarios a la hora de describir un modelo de tópicos, estos son *palabra, documento* y *colección*:

Definición 2.12. Una *palabra* es una unidad básica de información discreta que en este contexto se define como un elemento de un vocabulario indexado V. Para efectos del modelo, las palabras son representadas como vectores unitarios que tienen sólo una componente en 1 y todas las otras en 0. Así, la n-ésima palabra del vocabulario se define como un vector w de largo $|V|$ tal que $w^n = 1$ y $w^m \neq 0$ para todo $m \neq 0$

Definición 2.13. Un *documento* es una secuencia de palabras denotadas por $\mathbf{w} = (w_1, w_2, ..., w_N)$, donde w_n es la n-ésima palabra en la secuencia.

Definición 2.14. Un *corpus* es una colección de documentos denotada por:
$D = \{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_M\}$

La noción tras el modelo *LDA* es que sin importar la colección de documentos con la que se trabaje, cada uno de ellos puede presentar múltiples tópicos y además, estos tópicos no están relacionados entre si. Por ejemplo, si se tiene una colección de diarios de un día en específico, es posible asegurar que estos discutirán un conjunto de tópicos en común, que cada diario discutirá una serie de tópicos y que cada tópico discutido no tendrá relación con los otros.

Este modelo utiliza la noción de generación de documentos, tópicos y palabras a lo largo del tiempo, por lo cual se extraen las siguientes definiciones:

Definición 2.15. Un *tópico* es una distribución de probabilidad sobre un vocabulario fijo. Para una colección de documentos, se asume que estas distribuciones están dadas de manera previa a la generación de cualquier documento.

Por ejemplo, el tópico *fútbol* tiene palabras sobre este deporte como *arco, portero, defensa, delantero* con una alta probabilidad y el tópico *guerra* tiene palabras como *armamento, munición, muertos, heridos* con alta probabilidad.

Luego, el proceso de generación de cada documento en la colección se desarrolla como sigue:

1. Escoger una distribución aleatoria sobre los tópicos

2. Para cada palabra en el documento se tiene:

    a) Escoger un tópico de manera aleatoria en base a la distribución generada en el paso 1.

b) Escoger una palabra de manera aleatoria a partir de la distribución correspondiente al tópico sobre el vocabulario.

Tal como se puede observar en este sencillo algoritmo, el modelo LDA consiste de un modelo de probabilidades que busca organizar colecciones de documentos y, debido a su naturaleza generativa, es posible de asimilar al proceso de extraer esferas con múltiples características de una tómbola e ir deduciendo los conjuntos subyacentes que se encuentran en la colección a medida que cada esfera va saliendo a la luz, por ejemplo, es posible agrupar los objetos por color, por tamaño, por imperfecciones, etc.

Desde un punto de vista más formal, todo proceso generativo basado en probabilidades se basa en la existencia de variables no observables en la colección y por lo tanto, para obtener información sobre estas últimas es necesario inferir la distribución conjunta entre eventos conocidos y eventos latentes. Gracias a la estadística *bayesiana* es posible tener esta información a través del uso de distribuciones condicionales de estos eventos ocultos dado que ya se conocen las distribuciones de eventos observables. En el caso del modelo LDA, los eventos observables son la aparición de palabras en los documentos; y las variables ocultas son todas aquellas que caracterizan la estructura de tópicos de una colección de documentos.

Es decir, cada cada documento $\mathbf{w}$ en el corpus D se tiene:

1. Definir $N \sim Poisson(\xi)$
2. Definir la distribución $\theta \sim Dirichlet(\alpha)$
3. Para cada palabra w_n en $\mathbf{w}$
   a) Escoger un tópico $z_d \sim Multinomial(\theta)$
   b) Escoger una palabra w_d a partir de $p(w_n|z_n, \beta)$, la distribución multinomial de probabilidades condicionada sobre el tópico z_n.

En donde β es la matriz de probabilidades de aparición de una palabra en un tópico, donde $B_{ij} = p(w^j = 1|z^i = 1)$; θ_d es la distribución de tópicos para el documento d, con $\theta_{d,k}$ la probabilidad de que el tópico k se discuta en el documento d; z_d son las asociaciones de tópicos para el documento d con $z_{d,n}$ es el tópico asociado a la palabra n-ésima del documento d; y w_d son las palabras observadas en el documento d, donde $w_{d,n}$ es la palabra n-ésima del documento d.

A partir de esto, es posible definir el proceso generativo de documentos a través de la distribución conjunta de variables observables y no observables que sigue:

$$p\left(\beta_{1:K}, \theta_{1:D}, z_{1:D}, w_{1:D}\right) = \prod_{i=1}^{K} p(\beta_i) \prod_{d=1}^{D} p(\theta_d) \left(\prod_{n=1}^{N} p(z_{d,n}|\theta_d) p(w_{d,n|\beta_{1:K}, z_{d,n}}) \right) \tag{2.1}$$

Así mismo, es posible representar gráficamente este modelo probabilístico tal como se observa en la figura 2.7. Cada caja representa el proceso de elección de un elemento, con la caja exterior representando la aparición de documentos dentro del *corpus* y la caja interior como el proceso de selección de palabras y tópicos dentro de un documento.

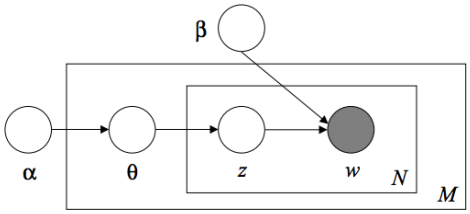

Figura 2.7: Representación gráfica del modelo LDA

Una vez definido el modelo que representa las relaciones entre los tópicos, los documentos y las palabras existentes en un *corpus*, para que este sea de utilidad es necesario calcular las distribuciones condicionales de la estructura de los tópicos dado la colección de documentos, esta distribución es lo que se llama como *posterior*. La definición de *posterior* se desprende de la ecuación 2.1 y se muestra a continuación:

$$p(\beta_{1:K}, \theta_{1,D}, z_{1:D}|w_{1:D}) = \frac{p\left(\beta_{1:K}, \theta_{1:D}, z_{1:D}, w_{1:D}\right)}{p(w_{1:D})} \tag{2.2}$$

Debido a la naturaleza de la ecuación 2.2, calcular el *posterior* es un problema NP completo, ya que la cantidad de estructuras de tópicos que se pueden encontrar en un *corpus* crece de manera exponencial y provoca que el denominador de esta ecuación sea imposible de calcular para problemas complejos. Así, es que para poder lograr descubrir la estructura de tópicos en un corpus lo suficientemente grande, es necesario utilizar algoritmos estadísticos que permitan estimar el *posterior*, entre estos algoritmos los más utilizados en la actualidad son los de muestro como el algoritmo de *Gibbs Sampling* [65, 69] los que permiten posteriormente inferir la estructura de tópicos en otros *corpus* existentes.

2.3.2. Correlated Topic Model

A diferencia del modelo LDA, el Correlated Topic Model usa una distribución de tópicos distinta, en la cual, se permite la existencia de covarianza entre las distintas componentes del modelo, la cual permite incluir la noción de que la presencia de un tópico latente puede estar directamente

relacionada con la de otro en el *corpus*. A modo de representación, en la figura 2.8 se muestra la representación gráfica del modelo probabilístico, para más información ver los detalles de este modelo en [8].

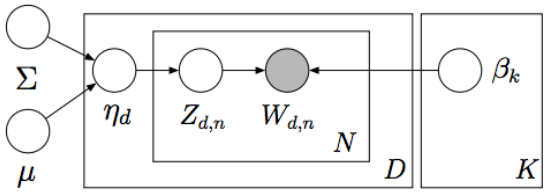

Figura 2.8: Representación gráfica del modelo CTM

2.3.3. Dynamic Topic Model

Si bien es una simplificación útil el considerar que palabras y documentos son intercambiables en las distribuciones de probabilidad presentadas anteriormente, para muchas colecciones de documentos esta simplificación suele ser errónea. En el caso de las colecciones de documentos como *artículos de revistas científicas, noticiarios* o cualquier otro repositorio de documentos que se enfoque en almacenar contenido que evoluciona a lo largo del tiempo, no sería posible descubrir cambios en tópicos entre dos periodos de tiempo distantes sin relacionar tópicos y documentos entre cada unidad de tiempo que se esté analizando. Es por esto que el modelo *Dynamic Topic Model* [9] propone un modelo con una serie de relaciones de tópicos y palabras entre distintos instantes discretos de tiempo, tal cómo se puede observar en la figura 2.9. Para más detalles en cómo estimar los parámetros de este modelo y sus distribuciones posteriores leer el paper *Dynamic Topic Models* [9] de Blei *et al.*.

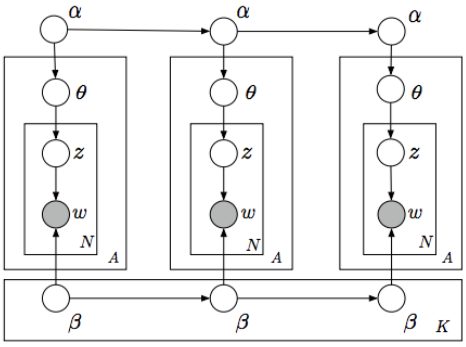

Figura 2.9: Representación gráfica del modelo DTM

2.4. Modelos de extracción de opiniones

En la actualidad, la mayor parte de las metodologías y algoritmos de recuperación de la información y de extracción del conocimiento sólo se enfocan en extraer información de datos duros o

documentos objetivos, sean estos generados por usuarios o por otro software. Sin embargo, con la llegada de la Web 2.0 y el empoderamiento del usuario en la generación de contenidos, estos últimos no sólo han generado conocimiento, también han aportado con su conocimiento subjetivo sobre hechos, tecnologías, productos y servicios entre otros. Este conocimiento subjetivo que ha aparecido en la web, ha provocado el nacimiento de un nuevo tipo de documento que va ganando cada vez importancia para las empresas, los documentos opinados, y como todo tipo de documento que requiere ser minado en busca de valor, se ha ido gestando una nueva corriente en el área de recuperación de la información, llamada *Web Opinion Mining*, la cual se se centra en extraer las opiniones de los documentos opinados que se encuentran en la web.

Una opinión se define como una creencia subjetiva sobre algún objeto, tema o situación en particular, resultado de una interpretación emocional de un hecho concreto o una característica del objeto en cuestión [18]. Por otro lado, una opinión se compone de: un receptor, aquel objeto, tema o situación en el cual se centra la opinión; una orientación, la cual señala la intención de la emoción que motiva la opinión y finalmente un emisor, aquel que consigna una opinión al público.

Al igual que todo modelo de recuperación de la información, los modelos de extracción de opiniones trabajan sobre documentos o colecciones de estos. Este tipo de modelos trabaja con *documentos opinados*, los cuales, de acuerdo a lo propuesto por Liu, B. en [40], pueden ser definidos como: todo documento que posea una o más oraciones opinada; o como aquel documento en el cual su autor desea expresar una opinión.

De acuerdo a Liu, B. [40], en el marco de la investigación de la Web, los usuarios pueden expresar sus opiniones sobre casi cualquier cosa (productos, servicios, experiencias, etc.) en una gran variedad de sitios web, lo que debido a la masividad de este medio de comunicación, convierte a la Web en una fuente virtualmente ilimitada de documentos opinados, los cuales poseen información relevante sobre la percepción de la gente frente a un siempre creciente universo productos y servicios.

aprovechar esta gran cantidad de documentos, y por consiguiente, busca analizar documentos, para determinar la existencia de opiniones en estos, y además,

El objetivo principal de cualquier modelo de extracción de opiniones es el determinar, a través del minado de documentos opinados, cuáles son las emociones y los sentimientos que guían la consignación de las creencias subjetivas (opiniones) en un documento [18].

2.4.1. Aplicaciones de los modelos de opiniones [55] [24]

En los último años los algoritmos de **opinion mining** han sido utilizados por una amplia gama de industrias, con un ejemplo importante a destacar el de la industria del *retail* debido a las variadas aplicaciones que estos han impulsado, como lo son las metodologías de *pricing* de productos y

servicios, el monitoreo de marca de una empresa y el análisis lanzamientos de nuevos productos.

A continuación se da una lista de otras aplicaciones en las cuales los algoritmos de opinion mining están siendo cada vez más y más utilizados:

1. **Industria del retail**: En esta área las aplicaciones que más destacan son aquellas que se enfocan en reviews de productos, entre las cuales las más tratadas en el ambiente académico son la detección de spam, generar resúmenes de las opiniones sobre productos y lograr identificar el nexo entre las opiniones y el precio que se le puede asignar a las características de un producto [2, 27, 31, 59]. Además, el minado de opiniones permite evaluar el impacto económico que tienen las reviews de un producto en las ventas de este.

2. **Sistemas de recomendación**: Las opiniones que un usuario emite en la Web permiten tener un mejor panorama de sus gustos, previniendo realizar sugerencias acorde a lo que el usuario espera adquirir, o incluso, realizar ofertas en busca de mejorar la percepción que este tiene sobre un producto o servicio en particular. En [19, 66, 67] señalan distintos acercamientos sobre el uso de opiniones para incluir nuevas funcionalidades a los sistemas de recomendación que ya existen.

3. **Inteligencia de negocios**: Es posible predecir el impacto sobre las ventas de una empresa que tendrá un producto a través de analizar las opiniones presentes en la web [41, 46].

4. **Política**: Una de las áreas en las que las emociones de la gente tiene mayor relevancia es la política, donde los candidatos tratan activamente de lograr empatía con la gente y provocar emociones positivas para ganar su voto, por lo tanto, el poder analizar las opiniones de la gente en las redes sociales puede ser una herramienta potente para decidir en qué y cómo realizar una campaña electoral. En [52, 62, 64] se muestran distintas aplicaciones que apuntan a tal objetivo.

5. **Marketing online**: El minado de opiniones también permite evaluar el lanzamiento de nuevos productos a través del análisis de reviews en la web. En [20, 30] se muestra cómo medir el impacto de campañas virales y cómo mejorar un sistema de avisaje online haciendo uso de opiniones.

6. **Análisis financiero a través de opiniones**: Múltiples intentos se han realizado con el objetivo de ver la correlación entre las opiniones presentes en las redes sociales y los precios de las acciones [25, 63].

2.4.2. Conceptos relevantes para modelos de opinion mining

Según lo expuesto por Liu, B. en [40] y un gran número de otros autores, todo modelo de *sentiment analysis* o de *opinion mining* deben incorporar las nociones de *objeto*, *emisor*, *polaridad* y *opinión* para que este sea capaz de interpretar a cabalidad lo que un documento opinado busca dar a conocer:

Definición 2.16. Objeto: Un *objeto* o es una entidad consistente de (T, A) con T el conjunto de componentes de o y A el conjunto de atributos que le pertenecen.

Definición 2.17. Emisor o Fuente de opinión: El *emisor* de una opinión es aquella persona u organización que la expresa a través de algún medio en particular.

Definición 2.18. Opinión: Una *opinión* sobre una característica f del objeto o es una visión, actitud o evaluación emocional por parte de un *emisor de opinión* sobre esté. En [40] se describen las emociones como ***sentimientos y pensamientos subjetivos***, y [58] señala que toda emoción es una combinación de seis emociones básicas: *amor, alegría, sorpresa, rabia, tristeza* y *temor*.

Definición 2.19. Orientación de una opinión: La *orientación* de una opinión sobre un objeto o indica en qué punto del continuo entre una opinión netamente negativa y una netamente positiva se encuentra el documento siendo analizando. Se dirá que una opinión es neutra si se sitúa al centro del espectro de polaridad. A lo largo de este trabajo también se hará referencia a la orientación de una opinión como la *polaridad* de esta. Ejemplos de opiniones con distintas polaridades son las que siguen:

- **Opinion positiva**: "*La película Pulp Fiction es la mejor de todos los tiempos*".
- **Opinion negativa**: "*La comida de perro me desagrada*".

Aún cuando toda opinión se ve guiada por una emoción en particular, la manera en que esta es dada a conocer por el emisor las divide en dos categorías: las opiniones *explícitas* son aquellas en que se se distingue la opinión en una frase subjetiva; y las *implícitas*, donde la opinión en cuestión es expresada a través del uso de una frase objetiva. Un ejemplo de opinión explícita es *"me encanta el sabor de este helado"* y de opinión implícita es *"la linterna explotó a la semana de haberla comprado"*.

Según lo descrito por [27] bajo su modelo de análisis basado en características, un objeto o está compuesto por un conjunto de características $F = o, f_1, f_2, ..., f_n$, donde cada característica f_i es definida en base a un conjunto de palabras $W_i = w_{i1}, w_{i2}, ..., w_{im}$, donde w_{ij} es una palabra sinónima de f_i. Así mismo, un documento d se considera como un documento opinado si contiene opiniones referentes a uno o más objetos $(o_1, o_2, ..., o_q)$ emitidas por uno o más emisores $(h_1, h_2, ..., h_p)$.

Es importante destacar que en modelo basado en características, cada opinión o_j presente en un documento opinado se enfoca en un subconjunto F_j de características y no sobre el objeto en su totalidad. Bajo este modelo no sólo es posible clasificar opiniones entre positivas y negativas, también es posible definir opiniones directas y comparativas como sigue:

- **Opinión directa**: Es aquella que hace referencia sólo sobre una característica f_jk de un objeto o_j. Esta es representada como $(o_j, f_{jk}, oo_{ijkl}, h_i, t_l)$, con oo_{ijkl} la orientación de la opinión descrita, h_i aquel que emite la opinión y, t_l el instante de tiempo cuando h_i emitió la opinión en cuestión.

- **Opinión comparativa**: Tal como su nombre lo indica, una opinión de este tipo busca establecer una relación de comparación entre dos o más objetos y además, la opinión de un emisor h_i sobre sobre un conjunto común de características o atributos entre todos objetos bajo análisis.

2.4.3. Algoritmos de Opinion Mining

Detección de opiniones en documentos

La mayoría de los algoritmos de minado de opiniones se basan en el supuesto de que el documento siendo analizado contiene opiniones. Sin embargo, esto no siempre se cumple por lo que determinar si un documento posee o no opiniones e identificar qué partes de estos documentos las contienen [16]. De acuerdo a Mihalcea*et al.* [44], este es un problema más complejo de resolver que el de detectar la polaridad de una opinión, por lo que se espera que la existencia de algoritmos eficientes en esta área impacte de manera positiva toda esta rama de investigación.

Comparación de objetos o características

Los modelos de opiniones enfocados en comparación de atributos buscan determinar qué motivación guía al individuo que ha emitido la opinión analizada, a preferir una característica sobre otra, un hecho sobre otro e inclusive un producto o servicio sobre otro. Estos modelos son ampliamente utilizados en la industria del retail, especialmente aquellos que no tienen tienda física y sólo tiendas en la Web, donde se dispone de grandes volúmenes de información sobre cada producto ofrecido en forma de documentos opinados, y en el mercado de servicios a través de encuestas de satisfacción y evaluación de calidad de servicio.

Durante el último tiempo, los modelos de extracción de opiniones a través de comparación han adquirido un uso práctico importante, ya que al ser combinados con modelos microeconómicos de valoración [2] son capaces de determinar el precio subjetivo de una característica de un producto en particular, o el valor que le asigna el comprador al hecho de elegir un producto o servicio por sobre otro.

Detección de orientación de opiniones

Los modelos de opiniones basados en polaridad, buscan determinar la predisposición de la entidad siendo analizada hacia el hecho o el objeto en cuestión. En general, un modelo de polaridad busca

determinar si las emociones que mueven a esta entidad son positivos o negativos, permitiendo tener sólo un atisbo de lo que realmente le motiva. Sin embargo, esta herramienta es comúnmente utilizada en conjunto con técnicas de *Business Intelligence* [23] para agregar una dimensión extra a los análisis de los resultados operacionales de la empresa, la cual permite considerar la polaridad de la apreciación del consumidor obtenida a través de estos modelos a la hora de tomar decisiones estratégicas.

En el marco de esta investigación se utilizará este tipo de modelos en el análisis de las opiniones vertidas en documentos de *microblogging*, ya que en el marco de detección de tendencias, se considera más útil el conocer qué postura tiene la gente frente a los productos y servicios, que el saber con detalle qué sentimiento alimenta la apreciación que tiene el individuo por estos, o qué ventajas posee el producto o servicio por sobre la competencia.

Debido a la naturaleza de los documentos de microblogging, los que se caracterizan por su largo limitado y su carácter informal, múltiples publicaciones tratan de resolver el problema de descubrir la polaridad de una opinión en textos cortos [45, 61, 70], sin embargo, no todos ellos obtienen buenos resultados al ser utilizados en textos cortos de *microblogging* y tal como se muestra en [36], destacan dos tipos de algoritmos, los con información rotulada y los no supervisados, de los cuales se da a conocer más detalles a continuación.

2.4.4. Revisión de algoritmos para detección de orientación de opiniones

Algoritmos de clasificación a través de aprendizaje supervisado

Los algoritmos de extracción de opiniones a través de Machine Learning (ML) se enfocan en el uso de clasificadores que a medida que estos son utilizados aprenden patrones que serán utilizados posteriormente para clasificar nuevos documentos. A continuación se dará a conocer el algoritmo más utilizado debido a su simplicidad, el clasificador de Naive-Bayes.

Un algoritmo de clasificación basado en Naive-Bayes funciona bajo la premisa de máximizar la probabilidad $\Pr(c \mid D)$, es decir, la probabilidad de que el documento D tenga orientación c.

Luego, la orientación c de un documento D se obtiene al resolver el siguiente problema de maximización: $arg\,\max_{c \in C}\{\Pr(c \mid D)\}$

Para resolver este problema se utiliza la regla de Bayes:

$$c_D = arg\,\max_{c \in C}\left\{\frac{\Pr(D \mid c) \cdot \Pr(c)}{\Pr(D)}\right\} \tag{2.3}$$

Debido a que sólo es necesario comparar entre elementos y no obtener un puntaje específico, es posible descartar el denominador de la ecuación 2.3. Además, el clasificador de Naive-Bayes asume independencia condicional entre todas las orientaciones, se puede decir que:

$$\Pr(D \mid c) = \prod_{i=1}^{m} \Pr(w_i \mid c) = \prod_{i=1}^{m} \frac{\#(w_i, c)}{\#(w_i)} \tag{2.4}$$

Con $\#(w_i, c)$ el número de veces que la palabra w_i se ha encontrado en documentos de orientación c en el conjunto de entrenamiento y $\#(w_i)$ el número de veces que la palabra w_i aparece en este último. Para evitar que existan probabilidades 0, se realiza un proceso llamado "suavización de Laplace" que consiste en lo siguiente:

$$\Pr(D \mid c) = \prod_{i=1}^{m} \frac{\#(w_i, c) + 1}{\#(w_i) + m} \tag{2.5}$$

Con estas ecuaciones basta resolver el problema de maximización planteado para obtener las probabilidades de pertenencia de cada documento a una orientación en particular.

En todo algoritmo de aprendizaje supervisado es importante definir que características de los documentos serán utilizadas para clasificar cada documento, entre estas, las más utilizadas son:

- *Presencia de términos*: En el área de la recuperación de la información se hace un uso extensivo del modelo *tf-idf*, sin embargo, tal como se da a conocer en [56], en el caso de los algoritmos de minado de opiniones el concepto de frecuencia no cobra mayor importancia y se ve relevado por la *presencia* de un término en un documento.

- *Partes del discurso*: Esta característica hace referencia al rol que cumple una palabra en particular dentro de un documento. Por ejemplo, en [49] se hace uso de palabras identificadas como adjetivos para clasificar documentos opinados, cuya hipótesis se ve corroborada en [26], donde se demuestra que existe una alta correlación entre la presencia de adjetivos en una oración y la subjetividad de esta.

- *Sintáxis*: La estructura sintáctica y gramatical de un documento ha sido utilizada en [43, 51] para tratar de identificar negaciones, sarcasmo y otras características de las opiniones que no son detectables a través del uso de las características previamente mencionadas.

Algoritmos de aprendizaje no supervisado

En general, los algoritmos de minado de opiniones basados en aprendizaje no supervisado hacen uso de la estructura de un texto y de la información existente sobre las partes del discurso para tratar de inferir la orientación de este. Un clásico ejemplo de algoritmo no supervisado es presentado en [68]:

1. Para comenzar, se proceden a recuperar del documento todas aquellas frases que contengan uno o más verbos y/o adjetivos. Para complementar la información representada por estos, es necesario contextualizarlos tal como se muestra en la tabla 2.1, donde se hace uso de tríos de palabras, donde siempre una de ellas es un adjetivo.

Primera Palabra	Segunda Palabra	Tercera palabra
Adjetivo	Sustantivo Plural o Singular	Palabra
Adverbio	Adjetivo	No Sustantivo Plural ni Singular
Adjetivo	Adjetivo	No Sustantivo Plural ni Singular
Adjetivo	Adjetivo	No Sustantivo Plural ni Singular
Sustantivo Plural o Singular	Adjetivo	No Sustantivo Plural ni Singular
Adverbio	Verbo	Palabra

Cuadro 2.1: Patrones de partes del discurso

2. Para cada una de las frases recuperadas en base a los patrones del paso anterior, se estima su polaridad a través de ecuación 2.7 haciendo uso la métrica de dependencia estadística *pointwise mutual information* (PMI) que se presenta en la ecuación 2.6

$$PMI(w_1, w_2) = log_2 \left(\frac{\Pr(w_1 \wedge w_2)}{\Pr(w_1) \Pr(w_2)} \right) \tag{2.6}$$

$$oo(frase) = PMI(frase, \text{“}excelente''\text{)} - PMI(frase, \text{“}pobre''\text{)} \tag{2.7}$$

3. Finalmente, el algoritmo calcula la polaridad *oo* promedio de todas las frases en el documento y lo clasifica dependiendo de si el promedio es positivo o negativo.

Lexicones de opinión

Un lexicón es un recurso de información rotulada que asocia palabras con polaridad de sentimientos. Dicho de otro modo, a cada palabra le asocia un valor en el continuo de polaridad de opiniones. El uso de los lexicones en algunos modelos de opinión mining se basan en la hipótesis de que una palabra puede ser considerada como una unidad fundamental de información sobre opinión, y por lo tanto puede dar indicios sobre la polaridad de un documento en su totalidad. Algoritmos de opinion mining que se basan en el uso de lexicones pueden ser encontrados en [14, 35, 53, 56], los cuales de acuerdo a lo mostrado por Kouloumpis *et al.* [36] pueden dar buenos resultados en el contexto de minado de opiniones desde documentos de *microblogging*.

SentiWordNet [22] es un recurso de información léxica disponible públicamente para ser usado en aplicaciones de Web Opinion Mining. Este repositorio léxico fue creado el año 2006 con el fin de ser

utilizado en múltiples aplicaciones de opinion mining y sentiment analysis, y una versión mejorada de este fue lanzada el año 2010 [3]. En este trabajo de investigación se hará uso de la versión 3.0 de SentiWordNet[10].

Cada palabra existente en el lexicón tiene asociado un puntaje, en el caso de *SentiWordNet* [53] se tiene que cada palabra tiene asociado dos puntajes:

$$\vec{w} = \langle w^n, w^p, w^o \rangle \tag{2.8}$$

Con $\vec{w}$ el vector rotulado de la palabra w, w^p el puntaje positivo de la palabra, w^n el puntaje negativo y w^o el puntaje objetivo de esta.

Además, cada palabra rotulada presente en *SentiWordNet* posee la siguiente restricción sobre sus componentes:

$$w^p + w^n + w^o = 1 \tag{2.9}$$

La ecuación 2.9 implica que una palabra está formada en su totalidad por una componente positiva, una negativa y otra objetiva.

Algoritmos basados en lexicones de opinión

Hay diversos enfoques de la extracción de opiniones a partir de lexicones de opinión, sin embargo, sólo se mencionarán los más utilizados a continuación:

- **Conteo de palabras**: En este tipo de algoritmos, para asignar un puntaje a un documento se parte por realizar un conteo de aquellas palabras con una connotación más negativa que positiva o viceversa. Así, una palabra será considerada negativa si su puntaje negativo es mayor a su puntaje positivo, y así mismo, una palabra será considera positiva si su puntaje positivo es mayor a su puntaje negativo. Luego, el puntaje positivo del documento es la cantidad de palabras con connotación positiva, y la misma lógica sigue en el caso del puntaje negativo.

- **Promedio de palabras**: En un algoritmo de promedio de palabras, el puntaje positivo y negativo de un documento es el promedio de estos valores para cada palabra en el documento que tenga algún grado de connotación, en el caso de *SentiWordNet*, es el promedio de todas aquellas palabras que cumplen la condición $w^n < 1$.

- **Máximo del documento**: En estos, el puntaje positivo y negativo de un documento es el máximo entre los puntajes de cada palabra. Así, el puntaje positivo del documento es el máximo entre todas las palabras, y lo mismo ocurre con el puntaje negativo.

[10]Disponible en `http://sentiwordnet.isti.cnr.it` al momento de la publicación de este documento.

Diversas mejoras se pueden realizar a estos algoritmos, en [54], sugieren modificadores para los puntajes iniciales de cada palabra en base al vecindario de palabras en el cual estas se encuentran. Entre estas mejoras se encuentran la detección de negaciones y capitalización, y la existencia de intensificadores y disminuidores.

2.5. Soluciones existentes para detección de tendencias

En el ámbito académico, múltiples investigaciones [1, 17, 42] han abordado la detección de tendencias en la Web, principalmente en las redes sociales, destacándose entre ellas dos tipos distintos, aquellas que tienen como objetivo detectar de manera temprana tópicos que serán tendencia en el corto plazo, y las que buscan detectar aquellos tópicos que están siendo tendencia y su presencia va en aumento a lo largo del tiempo. La limitación de este tipo de modelos es que no permiten monitorear la evolución de los tópicos o tendencias a lo largo del tiempo, y por lo tanto no permite obtener una correlación con la demanda de productos y servicios en un mercado en particular.

En aplicaciones comerciales, la plataforma web *NewsWhip*[11] ofrece prestaciones similares a las que busca proveer la plataforma de detección de tendencias, sin embargo, su enfoque es lograr ser un agregador de noticias con características sociales, como la medición de menciones en las redes sociales de una noticia en particular o el análisis de noticias de una empresa en particular en la Web. Además, *NewsWhip* ofrece la herramienta *Spike*, que permite a los generadores de contenido analizar cómo sus noticias se esparcen por la Web. Si bien este tipo de herramientas permiten un análisis similar al propuesto en este trabajo, su enfoque es principalmente el análisis de noticias en la Web, a diferencia de lo propuesto en este modelo que busca analizar tópicos tanto en la cantidad de cobertura noticiaria que tienen como en el sentimiento generado en las redes sociales por parte de ellos.

La empresa *Sysomos*[12] se enfoca en monitorear las redes sociales en búsqueda de información relevante para una empresa en particular, sin embargo, no hacen uso de la información presente en las noticias y no tienen como objetivo hacer un análisis extenso de las tendencias en la web, si no monitorear las conversaciones que se están realizando en las redes sociales.

Otra iniciativa que busca detectar tendencias en la Web es Google Trends, la cual toma un enfoque distinto a los ya mencionados al analizar el comportamiento de búsqueda de los usuarios de su motor de búsqueda, sin embargo, no hacen uso de los datos presentes en su red social Google+ para complementar las tendencias obtenidas con información sobre las opiniones de la gente sobre ellas.

[11] `http://www.newswhip.com/`
[12] `http://www.sysomos.com/`

A pesar de buscar herramientas de código abierto o propietario que permitieran el tipo de análisis deseado por Duam S.A., no fue posible encontrar alternativas disponibles, principalmente debido a que toda empresa que ofrece servicios similares poseen patentes sobre su tecnología y no tienen a disposición su plataforma para que sea utilizada por terceros para ofrecer servicios similares.

Capítulo 3

Detección de Tendencias en la Web

En el presente capítulo se dará a conocer el diseño general de la plataforma de detección de tendencias el cual se divide en tres etapas: definición de los requisitos de la plataforma, diseño de los módulos de recuperación de documentos e información; y finalmente, la herramienta de visualización de tendencias en base a la información recuperada.

La primera sección de este capítulo listará los requisitos que debe satisfacer la plataforma para poder cumplir los objetivos planteados anteriormente. Estos se dividen en múltiples ámbitos: requisitos de integración, aquellos que tienen como objetivo desarrollar una plataforma modular que permita mejoras y el uso de sus módulos de manera independiente; requisitos de usuario, que buscan definir la manera en que serán usadas las herramientas por el usuario y de que forma se recibirá información que este deba ingresar al sistema; y requisitos funcionales, los cuales apuntan a definir cómo deben ser diseñadas las componentes del sistema y las interacciones entre estas.

Por otro lado, se dará a conocer la solución propuesta para desarrollar la plataforma que permitirá llevar acabo este trabajo de investigación. Inicialmente se dará a conocer una descripción general de las componentes que alimentan el sistema, es decir, el módulo de minado de tópicos y el de minado de opiniones; seguido por delinear las interacciones entre estas y finalmente el módulo de detección de tendencias.

En la última sección de este capítulo se dará a conocer el producto final de esta tesis: una herramienta que permita visualizar información presente en la Web a través del cruce entre tópicos y opiniones, de manera tal que entregue indicios sobre las tendencias que se van generando en esta, señalando la opinión que tienen los usuarios sobre los tópicos que están en discusión y la presencia en los medios que estos han tenido a lo largo del tiempo.

3.1. Requisitos de la plataforma

En esta sección se definirán los requisitos que debe cumplir la plataforma de detección de tendencias a desarrollar en este trabajo de investigación.

3.1.1. Actores de la plataforma

Antes de definir los requisitos que debe satisfacer la plataforma, es necesario determinar qué actores se ven involucrados con el sistema y de que manera estos se relacionan con él, para así tener mayor claridad a la hora de definir y priorizar requisitos. Los actores que se ven involucrados en este sistema son:

- **Usuario**: este se relaciona con el sistema a través de la definición de fuentes a minar y además la visualización de la información recuperada por el sistema.
- **Fuente de documentos**: Aquel sitio web que es minado por el módulo de recuperación de documentos.
- **Red social**: Sitio web orientado a permitir que sus usuarios se comuniquen e interactúen entre si. Son una de las principales fuentes de información opinada en la Web.

3.1.2. Requisitos de integración

Los requisitos de integración son aquellos que apuntan a enmarcar el diseño de la plataforma de detección de tendencias con miras a la integración de todos los módulos que la componen de manera tal que no implique un acoplamiento excesivo entre estas. Así mismo, al reducir la cohesión entre las componentes se permite su uso de manera independiente si es que se desean utilizar otras fuentes de información para cada una de las etapas del proceso de detección de tendencias. Estos requisitos se listan a continuación:

- Cada módulo de este sistema debe ser capaz de interactuar con los otros de manera asíncrona, es decir, cada componente debe funcionar de manera independiente y no necesariamente secuencial con respecto a los demás.
- Cada módulo debe guardar todo documento que recolecte en un repositorio centralizado de almacenamiento de información, de preferencia una base de datos, junto con todos los otros datos o información que este extraiga a través de su ejecución como por ejemplo, tópicos para cada instante de tiempo, relaciones entre documentos y tópicos, relaciones entre tópicos a lo largo del tiempo, etc.

3.1.3. Requisitos de usuario

Son aquellos que buscan encausar el diseño de la plataforma hacia el uso de esta por parte de un usuario final. Estos buscan caracterizar la interacción entre este y la plataforma, y además, la manera en que la plataforma recibe los datos y de qué forma la información extraída es presentada al usuario. Los principales requisitos que caen dentro de esta categoría son:

- La información extraída desde la Web en relación a la detección de tendencias debe ser presentada de manera visual para que permita que el usuario sea capaz de interpretarla de manera simple e intuitiva.
- El usuario debe ser capaz de definir las fuentes sobre las cuales se realizará el análisis de tendencias, sin embargo, estas deben cumplir las restricciones que se darán a conocer en el capítulo siguiente que hacen referencia a la naturaleza y a la disponibilidad de las fuentes.
- Para el usuario debe ser transparente el trabajo de la plataforma una vez que esta esté realizando sus análisis y sólo debe interactuar con ella cuando desee agregar nuevas fuentes o analizar los resultados obtenidos.

3.1.4. Requisitos funcionales

Son los que definen como deben ser diseñadas las componentes del sistema y las interacciones entre estas para satisfacer los requisitos mencionados previamente. Los requerimientos funcionales extraídos a partir de los requisitos ya mencionados son:

- Cada uno de los módulos deben estar implementados como servicios que estén en ejecución permanentemente. Así, se busca que los análisis sean realizados periódicamente de manera automática y además que no ocurran pérdidas de información debido a que los algoritmos de recuperación de documentos (sean estos opinados o no) no hayan sido ejecutados de manera oportuna.
- Cada módulo debe guardar su información en una base de datos centralizada. Esto es necesario para permitir la existencia de dependencias entre las tablas que contendrán toda la información extraída, ayudando así a que la base de datos sea consistente.
- El módulo de recuperación de documentos debe realizar peticiones a las fuentes escogidas de manera periódica y no continuamente para evitar saturar sus servidores y por consiguiente tener restricciones de acceso a la información.

3.1.5. Requisitos no funcionales

Apuntan a cualquier cualidad que deba poseer la plataforma que no describa directamente una funcionalidad de esta.

1. **Tiempo de procesamiento**:

   El sistema debe poder procesar la información que este recibe dentro de un periodo de trabajo. Este requisito debe cumplirse tan sólo para un nicho de mercado en particular y una lista de fuentes fija en el tiempo.

2. **Orientación a Objetos**:

   El código que sustenta a la plataforma debe ser programado utilizando orientación a objetos con el fin de permitir la extensibilidad del sistema una vez que la versión inicial de este ya haya sido desarrollada.

3. **Asincronía entre lectura y escritura de datos**:

   El sistema debe permitir la lectura de la información extraída en cada módulo sin interferir con el proceso de detección de tendencias, así, es posible visualizar la información extraída sin importar que se estén llevando acabo los procesos de recuperación de la información dentro del sistema.

3.2. Plataforma de detección de tendencias

Tal como se mencionó al inicio del capítulo, para el desarrollo de una plataforma de detección de tendencias en la Web, es necesario minar tópicos desde fuentes de documentos como blogs o sitios de noticias y minar documentos opinados a partir de sitios de redes sociales. Además, es necesario procesar todo documento procesado para recuperar la información que se quiere obtener: en el caso de los artículos se busca obtener los tópicos que se discuten en ellos; y en el caso de los documentos opinados se buscar extraer la orientación de las opiniones que estos contienen.

3.2.1. Componentes de la plataforma de detección de tendencias

Los módulos de recuperación de la información que deben ser construidos para poder crear la plataforma de detección de tendencias son:

- **Crawler de artículos**: Crawler enfocado en recuperar documentos desde una serie de fuentes definida por el usuario. Estos documentos pueden estar enfocados en hechos, productos, individuos o prácticamente cualquier tema que pueda marcar tendencia. Este módulo será el

que recuperara los datos necesarios para alimentar el módulo de minado de tópicos descrito a continuación.

- **Módulo de minado de tópicos**: Utiliza como *input* un corpus de documentos ordenados cronológicamente según la fecha en que estos fueron publicados. Luego, se dedica a minar los documentos recuperados por el crawler extraer los tópicos que se discuten periodo a periodo, y analizar la evolución de los tópicos a lo largo del tiempo. Su *output* es una serie de tópicos que luego serán utilizados por el crawler de opiniones para recuperar documentos opinados desde redes sociales.

- **Crawler de opiniones**: Recibe una serie de tópicos a partir de los cuales debe buscar documentos en las redes sociales. Es el encargado de, dado un tópico, determinar qué consultas debe realizar para recuperar un conjunto de documentos opinados capaz de representar al tópico en cuestión.

- **Módulo de minado de opiniones**: Dado un conjunto de documentos opinados y de tópicos, este módulo cumple la función de asignar a cada uno de estos últimos un puntaje de opinión asociado con el periodo en el que se está corriendo el análisis. Así, este módulo es el que, para aquellos tópicos que abarcan múltiples periodos, determina los cambios en la percepción de la gente sobre estos.

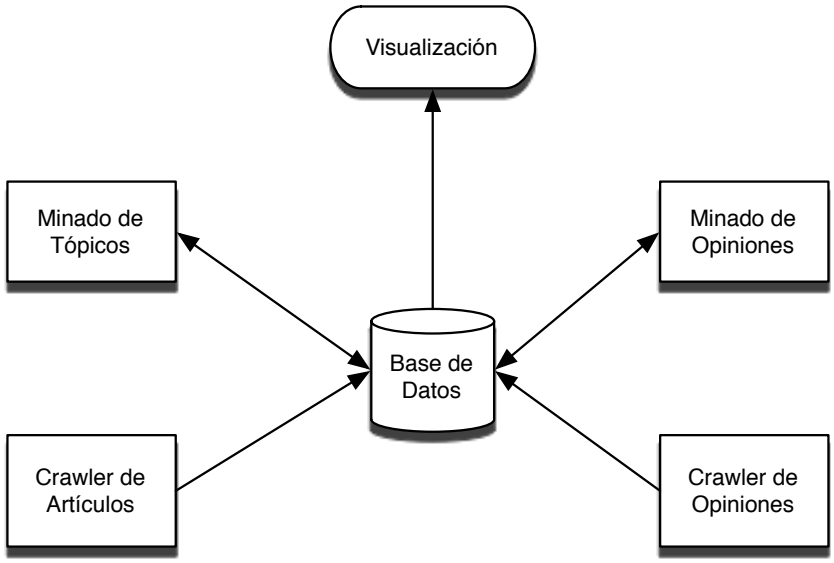

Figura 3.1: Diagrama de interacción entre componentes del sistema

Debido a que se busca crear una herramienta que permita visualizar las tendencias en los nichos de mercado que están bajo análisis, A esta lista de módulos que se deben construir se le deben sumar

dos componentes extras tal como se puede observar en la figura 3.1. Estas son:

- **Base de Datos**: Este módulo será el encargado de almacenar toda la información recopilada por los demás módulos de la plataforma. Sólo se utiliza como un repositorio de información y datos, no realiza ningún tipo de procesamiento sobre estos.
- **Visualización de Tendencias**: Una vez que todos los módulos hayan cumplido su función dentro de la plataforma, y cada documento esté asociado a un tópico opinado[1], el módulo de visualización de tendencias tendrá como objetivo el otorgar al usuario el acceso a una representación gráfica de toda la información recuperada, lo que permitirá detectar tendencias en el tiempo.

3.2.2. Arquitectura tecnológica

En cuanto la arquitectura física que sustenta la plataforma, se observa la arquitectura utilizada en la figura 3.2, la cual consiste de una capa de persistencia donde se encuentra la base de datos

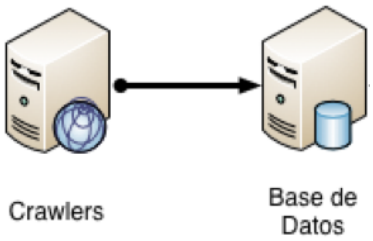

Figura 3.2: Arquitectura física para la plataforma de detección de tendencias

3.3. Visualización de tendencias

Tal como se ha enunciado a lo largo de este capítulo, la principal contribución de este trabajo de investigación es la creación de una metodología de detección de tendencias en la web, la cual es complementada con el desarrollo de una herramienta de visualización de la información extraída por cada uno de los módulos mencionados en la sección anterior. Un ejemplo de esta herramienta se presenta en la figura 3.3[2], donde las barras representan la cantidad de documentos en un periodo y la curva representa los puntajes de opinión correspondientes.

[1] Se define **tópico opinado** como aquel que tiene asociado un puntaje de opinión
[2] Fuente: Elaboración propia.

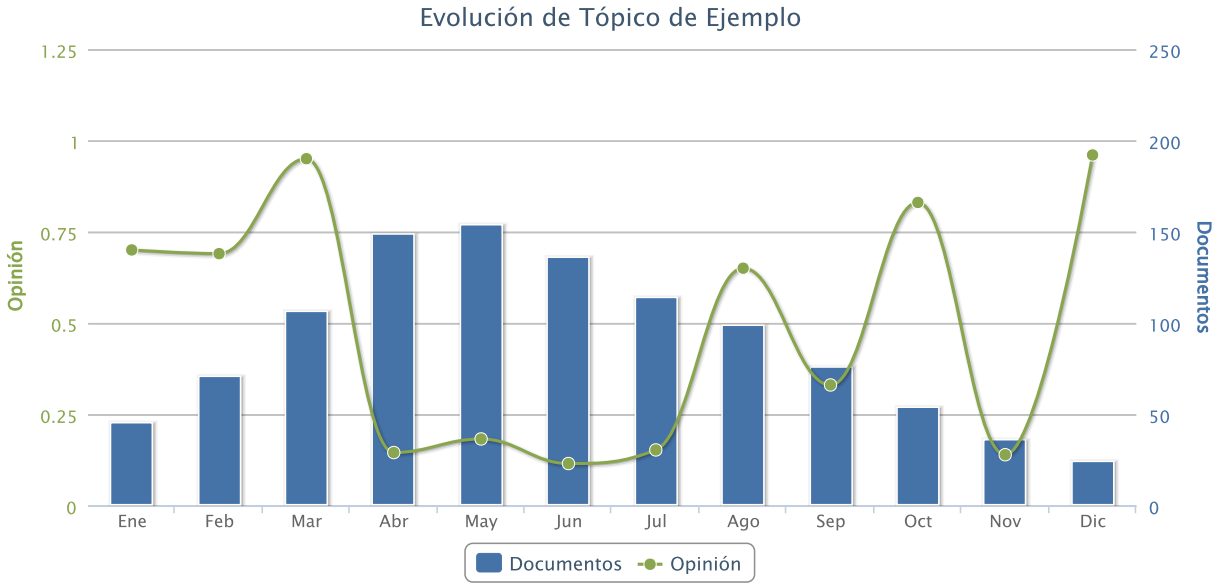

Figura 3.3: Ejemplo de gráfico por tópico.

Capítulo 4

Módulo de minado de tópicos

En este capítulo se dará a conocer la plataforma propuesta por esta tesis para la extracción de tópicos a partir de fuentes de noticias o artículos pertenecientes a blogs. Específicamente, se darán a conocer los requisitos pertinentes a este módulo, la manera en que se almacenarán los datos recopilados por esta plataforma y los algoritmos a utilizar.

Para comenzar, se darán a conocer cuáles son los requisitos que debe satisfacer tanto esta plataforma como los sitios que podrán ser procesados por ésta. Cada requerimiento que se describirá en esta sección será desde un punto de vista funcional, es decir, estarán enfocados en lo que debe lograr o poseer el módulo en cuestión, y no en *cómo* lograrlo desde el punto de vista del diseño de software.

Posteriormente, se dará a conocer el diseño bajo el cual se desarrollará esta plataforma, describiendo cada una de las componentes que forma parte de esta y además, detallando en qué manera cada una de ellas se relaciona con las demás. Finalmente, se realizará una descripción general sobre lineamientos que la arquitectura tecnológica que albergará esta solución en un ambiente de producción debe cumplir.

Para finalizar, se detallará cada uno de los procesos que ocurren en la plataforma de extracción de tópicos, es decir, el proceso de **recuperación de documentos**, el de **procesamiento de documentos y reducción dimensional**, **extracción de tópicos a partir de documentos previamente recopilados** y finalmente, el proceso que logrará determinar la evolución de los tópicos a lo largo del tiempo. Esta descripción en detalle será realizada desde un punto de vista teórico, mencionando los modelos y algoritmos a utilizar en cada caso.

4.1. Objetivo

Entre los tres módulos que componen la plataforma de detección de tendencias que se construyó en este trabajo de investigación, el pilar principal de ella es el módulo de detección de tópicos. Este debía cumplir con dos objetivos, por un lado es el encargado de recuperar a partir de una lista de fuentes preestablecida todos los documentos que estos provean, y además tiene como objetivo final el determinar un modelo de tópicos capaz determinar qué tópicos se discuten entre estos documentos para cada periodo de tiempo y la evolución de estos a lo largo del tiempo.

4.2. Requisitos funcionales de la plataforma y de fuentes a procesar

A la hora de definir cuáles serán los requisitos prioritarios de este módulo de la solución, es necesario hacer hincapié en dos áreas que son consideradas basales a la hora de realizar una solución acorde a lo definido en el capítulo 3, estas son la manera en que este módulo procesará los documentos recuperados y las características de los sitios que serán minados con el fin de obtener documentos que alimenten el algoritmo de extracción de tópicos.

4.2.1. Fuentes a procesar

Si bien no todas las entradas generadas a lo largo del tiempo por un sitio deben estar disponibles en todo instante de tiempo, sí es necesario que sea posible almacenar este historial de entradas, lo cual es factible si en un instante de tiempo determinado es posible acceder a todos los documentos generados hasta ese momento desde la última vez que se visitó el sitio. Así, si se cumple este requisito, es posible reconstruir el historial de entradas de una fuente a partir de una serie de capturas de ésta en una serie de intervalos de tiempo discretos. Luego, es posible aseverar que si se cumple este requisito y se recupera la información de una fuente F en los instantes $\{t_i\}_{i\in\mathbb{N}}$, los documentos D^F pertenecientes a esta fuente se pueden definir como sigue:

$$D^F = \bigcup_i \{d^{F^{t_i}}\} \tag{4.1}$$

Es decir, es posible obtener el conjunto completo de documentos generados por la fuente F desde el instante t_0 sin necesidad de realizar el proceso de recuperación de documentos continuamente, si no sólo visitando la fuente en instantes equidistantes de tiempo, minimizando los recursos utilizados para su recuperación y el impacto sobre la fuente de contenido en caso de que sea utilizada en este proyecto de investigación.

Además, para facilitar el proceso de extracción de documentos a partir de las fuentes ya escogidas, junto con lograr extraer *metadata* que no se encuentra directamente en el documento a procesar,

es deseable que las fuentes escogidas sean sindicables, en particular, se espera que el contenido esté disponible en los formatos *RSS*[1] o *Atom*. Este requisito no reduce significativamente el universo de fuentes factibles a minar, ya que en la actualidad la mayor parte de las fuentes de noticias y blogs, y la totalidad de los más relevantes en cada área, proveen sus artículos a disposición en alguno de estos formatos.

Finalmente, todas las fuentes a procesar deben estar en el mismo idioma ya que aún cuando se pueden tener múltiples instancias de la plataforma extrayendo documentos de distintos idiomas, los modelos escogidos para trabajar en esta plataforma sólo son capaces de detectar la estructura de tópicos existente en una colección compuesta de documentos del mismo idioma.

4.2.2. Requisitos de procesamiento de información

Finalmente, es necesario considerar requisitos desde el punto de vista desde el procesamiento de la información, los cuales apuntan a definir qué es lo que se debe almacenar y qué procesos deben ser realizados para obtener resultados a partir de los datos, sean estos documentos o de cualquier otra índole.

Es de suma importancia almacenar todos los documentos que el crawler recupere de manera íntegra sin alterar el contenido de estos, para así poder realizar otros tipos de análisis posteriormente si así se quiere. Por otro lado, es deseable la capacidad de almacenar la *metadata* asociada a cada documento como la fecha de publicación, la categoría bajo la que este fue publicado en su fuente, autor, etc. ya que estos pueden ser utilizados como apoyo en caso de que se quiera realizar análisis sobre el tipo de documentos que se recuperan.

4.3. Descripción de la solución

En esta sección se da a conocer la primera componente de la solución propuesta, es decir, el módulo de recuperación de documentos y extracción de tópicos, el cual se dedica a analizar múltiples fuentes de información con el fin de describir qué es lo que está siendo publicado por este conjunto de fuentes. Este módulo se divide en múltiples componentes: recuperación de documentos, procesamiento de documentos y reducción dimensional, y el módulo de extracción de tópicos. En la figura 4.1 se observa un diagrama de la interacción entre estas componentes:

[1]Really Simple Syndication. `http://www.rssboard.org/rss-specification`

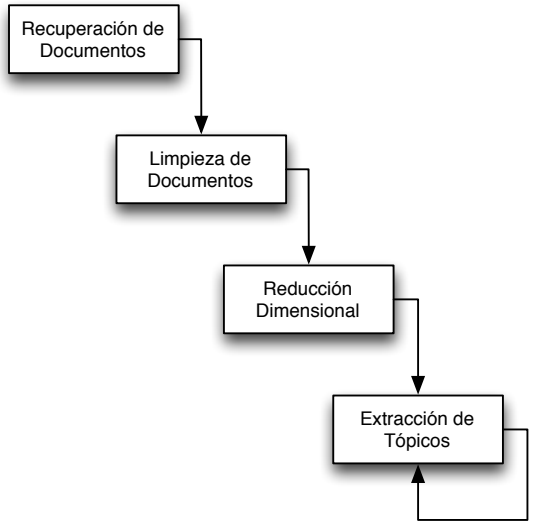

Figura 4.1: Diseño del módulo de minado de tópicos

4.4. Recuperación de documentos

Para dar a conocer la solución que se utilizará para la recuperación de documentos, primero es necesario dar a conocer las definiciones con las que se trabajará en ella para posteriormente definir el algoritmo de recuperación de documentos y la manera en que estos serán almacenados para su uso.

Un *feed* F se define como una fuente de documentos d ordenados de manera cronológica y amparados bajo una misma temática. Además, se dice que un feed F es sindicable si se dispone de un punto de entrada donde se expongan los documentos pertenecientes a la fuente, en particular, todo feed que disponga sus documentos a través de los protocolos *RSS* o *Atom* se considera sindicable.

Una característica importante de la mayoría de los feeds sindicables, es que los documentos expuestos en el punto de entrada son dependientes del instante en el que se le visita, lo que significa que el conjunto de documentos $\{d_i^F\}_{i\in\mathbb{N}}$ presente al recuperar la lista de documentos desde la fuente F depende directamente del instante t en el cual se realice esta revisión, por lo tanto, se define $\{d_i^{F^t}\}_{i\in\mathbb{N}}$ como la lista de documentos disponibles en una fuente F en un instante t. Sea $\{F_i\}_{i\in\mathbb{N}}$ la lista de *feeds* que utilizará el *crawler* para recuperar documentos a partir de sitios previamente definidos.

En base a lo anterior, es posible definir un algoritmo de recuperación de documentos a partir de una lista de fuentes $\{F_i\}_{i\in\mathbb{N}}$ sindicables (en el caso de este trabajo de investigación en formato *RSS* o *Atom*) de la siguiente manera:

Algoritmo 4.4.1: Recuperación de documentos

Data: $\{F_i\}_{i\in\mathbb{N}}$, t

Result: $\bigcup_i \{d_j^{F_i^t}\}_{j\in\mathbb{N}}$

```
1 forall feed in {F_i}_{i∈ℕ} do
2     XML = retrieveXML(F_i);
3     documents = [];
4     forall d in parseXML(XML) do
5         documents = documents ⋃ d;
6 return documents;
```

Donde cada documento d es una tupla con la siguiente información:

- Fuente F: sitio desde donde se obtuvo este documento.
- Tiempo t: donde la publicación fue creada en la fuente F.
- Contenido c: Todo el contenido textual del documento.
- URL h: identificador único que identifica la ubicación de este documento en el servidor desde donde se recuperaron los documentos.
- Metadata $\mathcal{M}$: El conjunto $\mathcal{M}$ involucra toda aquella información relevante al documento que no fue considerada en el modelamiento previo de un documento. Ejemplos de información que cae en este conjunto son los *tags*, *categorías*, *fuente*, etc.

El algoritmo 4.4.1 consiste en el proceso de recorrer la lista de fuentes disponibles una por una, y para cada fuente F, acceder al punto de entrada disponible, recuperar la lista de documentos $\{d_i^{F^t}\}_{i\in\mathbb{N}}$ asociados a la fuente F en el instante t, procesar la entrada asociada a cada documento para finalmente almacenarla junto con toda la metadata disponible para el documento d. Sin embargo, este algoritmo sólo visita cada fuente una única vez, por lo que debe ser extendido con el fin de tener un algoritmo capaz de reconstruir para cada fuente F, el conjunto completo de documentos desde el momento inicial en que se inició el análisis de esta.

Para extender el algoritmo presentado con el fin de que la plataforma sea capaz de obtener los documentos necesarios para la detección de tendencias, es necesario ejecutar el algoritmo 4.4.1 de manera periódica, ya que así será posible capturar la totalidad del universo de documentos presente en todas las fuentes disponibles. Si consideramos t_F como el tiempo en que un artículo permanece en

una fuente F, podemos calcular el tiempo máximo entre ejecuciones del algoritmo, el cual se define en la ecuación 4.2:

$$\dot{T} = min(t_F) \tag{4.2}$$

Si el algoritmo 4.4.1 es ejecutado con una frecuencia de $\frac{1}{\dot{T}}$, deben cumplirse dos supuestos para que todos los documentos en el historial de cada fuente sean recuperados con éxito. El primer supuesto, consiste en que la ejecución total de una iteración del algoritmo, es decir, recorrer todas las fuentes y recuperar los documentos disponibles, tome menos que $\dot{T}$ y además, que cada fuente esté disponible para la recuperación de sus documentos en cada momento que se ejecute el algoritmo.

Si bien estos parecen supuestos que pueden mermar la utilidad del crawler de artículos, en la práctica no impondrán ningún impedimento para este, ya que los tiempos de caducidad de los artículos suelen ser superiores a un día, tiempo suficiente para minar cientos de fuentes. Por otro lado, aún cuando los instantes de tiempo en los cuales un sitio puede estar inalcanzable varían en frecuencia, las causas principales están bien definidas:

- Servidor no disponible: En caso de que el servidor no se encuentre disponible, tampoco lo estará para aquel usuario que publica las entradas en él, por lo que no se perderá ningún artículo publicado.
- Servidor inalcanzable desde el servidor de procesamiento: En la mayoría de los casos, la duración de los momentos en los cuales el servidor objetivo se encontrará inalcanzable será menor a $\dot{T}$, y en caso contrario, la cantidad de noticias no recolectadas por el crawler está dentro de un rango aceptable para permitir el correcto funcionamiento del algoritmo de detección de tendencias.

Dado lo anterior, se puede considerar que el crawler recuperará todos los documentos necesarios para que sea posible reconstruir el historial de documentos publicados por todas las fuentes contenidas en la lista a utilizar.

4.5. Procesamiento de documentos y reducción dimensional

Luego de que los documentos han sido recuperados por el algoritmo de crawling detallado en la sección anterior, es necesario procesarlos para que sean utilizados sin problemas por el modelo de tópicos que será planteado en la sección siguiente. A continuación se detalla el proceso de reducción dimensional que se realizará a todos los documentos recuperados de las fuentes seleccionadas de antemano. Para realizar este proceso sólo se utilizará el contenido de cada documento, que será

denotado como d^C.

En todo proceso de extracción de conocimiento a partir de texto y documentos, es necesario realizar un proceso de reducción dimensional para evitar ruido asociado a las diferentes variaciones de una misma raíz semántica o de palabras que no aportan valor, desde un punto de vista semántico, al documento. En general, se ha demostrado que realizar estos procesos sobre cada documento en un corpus, logra aumentar la precisión de los algoritmos y por consiguiente, obtener mejores resultados en conjunto con otros algoritmos, como lo son los algoritmos de opinion mining que serán utilizados en el módulo de extracción de opiniones.

Para cada corpus a utilizar, es decir, para cada conjunto de documentos asociado a un intervalo de tiempo en particular, se busca reducir la cantidad de ruido semántico que presenta cada documento con el fin de poder obtener un modelo de tópicos en el tiempo que represente de mejor manera la información presente en la blogosfera o cualquier subconjunto de fuentes que se esté analizando.

Dado lo anterior, es posible definir el algoritmo de reducción dimensional que se utilizará sobre cada documento como sigue:

Algoritmo 4.5.1: Reducción dimensional de documentos

Data: $D = \bigcup_i \{d_i^C\}_{i=1...\mathbb{N}}$, SW

Result: $\bar{D} = \bigcup_i \{d^{\bar{C}}{}_i\}_{i=1...\mathbb{N}}$

```
1 for d^C inD do
2     d^C ← stemming(d^C);
3     d^C ← removeStopWords(d^C, SW);
4 return ⋃_i {d^C̄_i}_{i=1...N}
```

Tal como se puede observar en el pseudo-código del algoritmo 4.5.1, los dos métodos de reducción dimensional que se utilizarán para la limpieza de los corpus a utilizar para la detección de tópicos son el de **stemming** y el de remoción de **stopwords**. El orden de estos no es conmutable ya que realizar *stemming* antes de remover *stopwords* permite retirar de los documentos palabras que de otra manera no serían consideradas como tales y que sólo agregan ruido a los modelos de extracción de tópicos.

En las subsecciones presentes a continuación se describirán ambos métodos de reducción dimensional, para obtener una descripción a fondo de estos, referirse al capítulo 2.

4.5.1. Stemming

El método *stemming* a utilizar es el algoritmo de Porter que se describe en profundidad en el capítulo 2, el cual consiste en recorrer un documento y aplicar una serie de reglas de transformación a las palabras que lo componen de manera tal que el nuevo documento sólo posea los *stems* o *lemas* del documento original. De esta forma, se espera que palabras como *perro, perrito* sean todas reducidas a *perro*, aumentando así la importancia de esta última dentro del documento y por consiguiente, dentro del corpus, lo que permite que el modelo de tópicos a utilizar represente de mejor manera la importancia de este concepto dentro del modelo estadístico que se utilizará para representar las relaciones latentes entre cada documento, tópico y palabra que se esté analizando.

4.5.2. Remoción de Stopwords

Las stopwords, tal como se describen en el capítulo 2 son palabras cuyo valor desde el punto de vista de extracción de conocimiento es estadísticamente es nulo. Si bien una stopword con una baja frecuencia de aparición en un documento no provoca mayores problemas debido al bajo ruido que esta causa, en el caso de que esta sí presente una alta frecuencia de aparición en un documento, el ruido estadístico que causa pasa a ser significativo y por lo tanto es recomendable retirar la stopword del documento. Para evitar este tipo de situaciones, se removerán todas las stopwords que se encuentren en los documentos utilizando la metodología presentada en el capítulo 2.

4.6. Extracción de tópicos

El objetivo de esta etapa es obtener la estructura de tópicos que se encuentra en una serie de documentos $\{d_i\}_{i\in\mathbb{N}}$ ordenados cronológicamente. En particular, se desea determinar qué tópicos se manifiestan a través del tiempo, y cómo evolucionan a lo largo de este.

Para la extracción de tópicos se utilizará el modelo LDA detallado en el capítulo 2, el cual busca descubrir la estructura de tópicos que conecta a un conjunto de documentos ya existente. En particular, este modelo logra capturar las relaciones entre documentos, tópicos y palabras a través de un modelo estadístico que define las distribuciones de probabilidades existentes para cada una de estas relaciones.

Para obtener el modelo de tópicos asociado a un conjunto de fuentes F a lo largo de una cantidad fija de periodos de tiempo, se utilizará una metodología iterativa. Para cada periodo t_i dentro del conjunto $\{t_i\}_{i\in\mathbb{N}}$, se considera todo documento recuperado en los dos periodos anteriores t_{i-1}, t_{i-2} y se entrena un modelo LDA con estos. Luego, para los documentos del periodo t, con el modelo LDA obtenido en el paso anterior, se realiza inferencia sobre el corpus para descubrir el modelo de tópicos

subyacente en estos. Esta metodología se puede ver descrita de manera algorítmica a continuación.

Algoritmo 4.6.1: Extracción de tópicos desde un corpus ordenado cronológicamente

Data: $t, \bar{D}$

Result: $\sigma^*_{\mathcal{C}}$

1 $\tilde{D} := \bar{D}_{t-1} \bigcup \bar{D}_{t-2}$;

2 $M := LDA(\tilde{D})$ entrenar un modelo LDA en base a los documentos de $\tilde{D}$;

3 $\mathcal{T}_t := Inferir(M, \bar{D}_t)$ inferir los tópicos de $\bar{D}_t$ utilizando el modelo LDA M;

4 **return** $\mathcal{T}_t$;

La estructura de tópicos para los documentos presentes en el periodo t, denotada $\mathcal{T}_t$, está compuesta por cada tópico τ extraído en el periodo t, y para cada uno de ellos se posee la siguiente información:

- $\Pr(\tau \mid d)$: Probabilidad de que un documento d trate sobre el tópico τ. Esta probabilidad es el puntaje que el modelo le asigna a un documento al hecho de pertenecer a un tópico en particular.
- ϕ^{τ}: La distribución de probabilidades que relaciona las palabras con los tópicos compuesta por $\Pr(\tau \mid w) = \phi^{\tau}_w$, es decir, la probabilidad de que la palabra w esté describiendo al tópico τ.

Esta metodología crea un modelo LDA para cada período que se esté analizando lo que a priori no permite detectar la evolución de tópicos en el tiempo, sin embargo, se puede adecuar esta metodología para detectar la evolución de tópicos a lo largo del tiempo, ya que al entrenar el modelo con los sucesos que han ocurrido en las semanas previas, se espera que el modelo sea capaz de enlazar los tópicos entre las distintas semanas que este sea utilizado, permitiendo así analizar la evolución de un tópico en el tiempo.

Una de las limitaciones de esta metodología es que no permite descubrir nuevos tópicos apenas aparezcan, ya que para que estos sean detectados debieron existir al menos en alguno de los períodos anteriores al actual, lo que impide el uso de esta plataforma como sistema de alerta temprana. Esta limitación no afecta a la plataforma de detección de tendencias, ya que este análisis no se efectúa en el mismo período sobre el que se está aplicando el modelo si no que sobre un conjunto de estos.

4.7. Evolución de un tópico a lo largo del tiempo

Tal como se mencionó en la sección anterior, la metodología propuesta hasta este punto no permite analizar la evolución de un tópico a lo largo del tiempo de manera no supervisada ya

que requeriría conectar la información entre periodos de manera manual. Por lo tanto, para que esta metodología sea capaz de detectar la evolución de tópicos a lo largo del tiempo, es necesario modificar el algoritmo de extracción de tópicos de manera tal que sea capaz de enlazar tópicos entre periodos contiguos. Para ello, se define una métrica de cercanía de tópicos en base a la información extraída del modelo LDA en la ecuación 4.3.

$$\theta(\tau, \tau') = \sum_{w_i \in \vec{w}_\tau} \sum_{w_j \in \vec{w}_{\tau'}} w_i - w_j \tag{4.3}$$

Así, para cada tópico τ extraído en el periodo t, se calcula la distancia de este con todo tópico τ' de los dos periodos anteriores $t-1$ y $t-2$, y si esta está bajo un valor umbral ρ, se incluye τ' entre los tópicos que están relacionados con τ. Finalmente, aquel tópico τ' con menor distancia a τ se define como el tópico padre de este último. Este método se explica a continuación:

Algoritmo 4.7.1: Evolución de un tópico a lo largo del tiempo

Data: $\tau, \mathcal{T}_{t-1,t-2}$

```
buff = [];
for τ' ∈ T_{t-1,t-2} do
    if θ(τ, τ') < ρ then
        buff = buff ⋃ {(τ', θ_{τ,τ'})}
precursor(τ) = min(buff, θ_{i,j})el tópico con menor distanciaθ_{τ,τ'};
```

En caso de que ningún tópico τ' logré una distancia con τ menor a ρ entonces se considerará que el tópico no tiene precursor en las semanas anteriores y se considerará como un tópico completamente nuevo.

Una vez que este proceso haya sido ejecutado para cada uno de los tópicos extraídos en el periodo t, se procederá a realizar el minado de opiniones que se describirá en el siguiente capítulo.

4.8. Modelo de Datos

En esta sección se presenta una propuesta de modelo de datos para almacenar toda la información recopilada a lo largo de todo el proceso de recuperación de documentos y extracción de tópicos. Este modelo de datos será el encargado de almacenar las noticias, los tópicos y las relaciones entre estos.

Para cada noticia se deben almacenar: la fecha de publicación, la fuente, la *url* del *permalink*[2], el título, el contenido sin procesar, la fecha en la cual fue añadida a la base de datos y toda la metadata

[2]Un *permalink* es una url de enlace al artículo que funciona bajo la premisa de dar un punto de acceso imperecedero que sobreviva a cualquier cambio que se realice en la estructura del sitio

asociada al documento. En este caso, la *metadata* se almacenará serializada para permitir la inclusión posterior de información, a medida que nuevas fuentes de noticias traigan nueva información que no se tenía considerada antes.

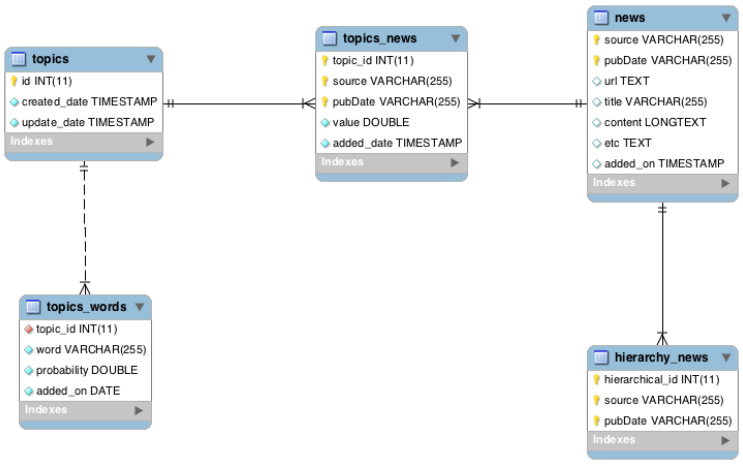

Figura 4.2: Modelo de datos para la recuperación de noticias y la extracción de tópicos

Por otro lado, para los tópicos es necesario que en la base de datos se encuentre información sobre: fecha de creación, fecha de actualización, palabras que identifican al tópico en un periodo en particular, y la probabilidad de que el tópico contenga la palabra mencionada en el periodo en el cual se realiza la asociación.

Además, es necesario tener la relación entre los tópicos y las noticias desde las cuales se extraen. En esta relación se debe considerar la fecha de asociación y el puntaje que el modelo de extracción de tópicos le da a la pertenencia de un documento d_i en relación al tópico τ_j correspondiente, es decir $\Pr(\tau_j \mid d_i)$.

Finalmente, existe una tabla extra en caso de que se quiera incluir posteriormente el uso de jerarquías para la clasificación de las noticias recuperadas, la cual contiene referencias a la llave primaria de la tabla que contiene la información sobre la taxonomía en cuestión y además, la referencia a la llave primaria del artículo. Esto es útil en caso de querer no sólo clasificar las noticias por tópicos, si no dar una clasificación especial de las noticias recuperadas que se sepa de antemano que facilita la comprensión de la información extraída.

Capítulo 5

Módulo de minado de opiniones

Para minar la Web en busca de patrones para la detección de tendencias, no es suficiente con sólo monitorear constantemente la información vertida en las fuentes de documentos tales como sitios de noticias y blogs, también, tal como se mencionó en el capítulo 1, es necesario conocer qué es lo que opinan los usuarios de la Web sobre los tópicos discutidos en ellas a través de las redes sociales.

Para poder detectar tendencias a través de información públicamente disponible en la Web no sólo es necesario conocer qué temas están siendo tratados en la blogosfera o en sitios de noticias, también es crucial comprender cómo reaccionan las personas a estos hechos. Hoy en día, bajo el alero del explosivo nacimiento de sitios que incorporan técnicas y tecnología relacionada a la Web 2.0, y el cambio de enfoque desde una web donde los usuarios sólo eran receptores de contenido a una web donde ellos son capaces de generar información, ha permitido que los usuarios de estos sitios web no sólo se dediquen a publicar artículos e información que recuperan desde otras fuentes, también poseen el espacio y la capacidad de expresar su opinión sobre todo lo que acontece en su día a día, sea esto su experiencia luego de consumir un servicio en alguna tienda en particular, su estado anímico en base a los sucesos que le acontecen y además qué piensan en relación a información publicada en fuentes presentes en la web, sean estas fuentes de noticias, artículos u otros.

Un tópico en la web puede ser considerado una tendencia si es que es publicado en muchos sitios a lo largo de varios periodos, sin embargo, sin incluir una componente que incluya la opinión de los usuarios, no es posible clasificar esta tendencia más allá de realizar una clasificación de acuerdo al impacto que el tópico en cuestión tiene en el universo de fuentes con el que se está trabajando. Por lo tanto, para obtener un sistema de detección de tendencias que sea capaz de diferenciar entre aquellas que la gente valora, desprecia o es indiferente, es necesario agregar una dimensión que tenga en cuenta la opinión de los usuarios de las redes sociales, permitiendo así obtener una mejor visión de cómo de comporta una tendencia y el impacto que tiene esta en la sociedad.

En base a lo anterior, se puede concluir que el módulo de web opinion mining tiene como ob-

jetivo general el detectar las opiniones expresadas en las redes sociales por parte de sus usuarios sobre algún tópico en particular. Para cumplir con este objetivo general se deben cumplir ciertos objetivos específicos, el primero de estos es el poder recuperar, dado un conjunto de palabras, documentos opinados asociados a ellas; y el segundo consiste en extraer un puntaje de opinión sobre cada documento recuperado.

En las secciones que se presentan a continuación se detallarán los algoritmos y metodologías que se utilizarán en cada una de las etapas del módulo de extracción de opiniones: recuperación de documentos opinados y la extracción de opiniones desde documentos. Además, se describirá la fuente de información opinada que será utilizada para el análisis, junto con una metodología para extraer consultas de búsqueda en base a un tópico que serán utilizadas para poder recuperar documentos opinados desde la fuente de documentos previamente escogida.

5.1. Fuentes de documentos opinados

A la hora de obtener documentos opinados en la web, es necesario tener una fuente que provea estos documentos para su recuperación y posterior procesamiento. En la actualidad, las principales fuentes de este tipo de documentos son los sitios de redes sociales como *Facebook* y los sitios de microblogging como *Twitter*, entre los ucales abarcan la mayoría de las opiniones que los usuarios de la web consigan en ella hoy en día.

Para que una fuente de documentos opinados sea factible de ser utilizada en una aplicación de detección de tendencias en la web, esta debe poseer al menos las siguientes características:

- Debe abarcar un amplio espectro de temas, desde el deporte hasta las ciencias, pasando por la educación, el bienestar social, la religión y cualquier otro tema que sea susceptible de querer analizar con un modelo de detección de tendencias.

- Un documento escogido arbitrariamente a partir de la colección, debe tener una alta probabilidad de ser capaz de reflejar la opinión que posee el emisor del documento sobre un hecho o una entidad en particular.

- La información debe estar etiquetada con atributos de tiempo y espacio, ya que se quiere analizar la evolución de la percepción de los usuarios del servicio con respecto a una entidad a lo largo del tiempo. A futuro es deseable tener la información necesaria para realizar análisis temporo-espaciales de las opiniones recolectadas, para así acotar la detección de tendencias a un área geográfica en particular.

- Cada opinión emitida por un usuario de esta red de opiniones debe ser de alto impacto, es decir, visible por una gran cantidad de usuarios estén inscritos para utilizar el servicio o no. Si bien es imposible asegurar un alto impacto efectivo, sólo basta con que el impacto de una opinión sea alto de manera potencial, es decir, la cantidad de usuarios que potencialmente podrían tener acceso a esta información debe ser alta y no necesariamente aquellos que efectivamente acceden a ella.

- Es una característica deseable, pero no estrictamente necesaria, el que los mensajes se enfoquen sólo en dar una opinión sobre una entidad por vez. Una de las pocas maneras de asegurar que esto se cumpla, es reducir la cantidad de información contenida en un documento de la colección, por ende, es deseable que los mensajes publicados sean de baja extensión.

En la actualidad, la mayor fuente de documentos opinados a nivel mundial de libre disposición y que cumple con todas las características ya mencionadas es *Twitter*, empresa norteamericana que ofrece el servicio de *microblogging*. De acuerdo a un reporte del área de ingeniería de *Twitter*, al 15 de Julio del año 2011, el servicio despacha más de 350.000 millones de *tweets*[1] cada día [21], con presencia en la mayor parte del globo, lo que lo hace un candidato excelente para ser minado en busca de opiniones sobre cualquier temática que se desee analizar.

Si bien en el último tiempo estudios han demostrado que tan sólo el 10 % de los usuarios de esta red social contribuyen con más del 90 % del contenido generado [7], esto se debe principalmente a que el 10 % de los usuarios son en su gran mayoría líderes de opinión dentro de este ecosistema, y otro puñado de usuarios son distribuidores de contenido no opinado, principalmente avisos de empresas que utilizan *Twitter* como un medio de comunicación efectivo con sus cliente y consumidores [72], lo que no afecta la utilidad de esta plataforma como fuente de documentos opinados.

Por lo expuesto anteriormente, este trabajo de investigación se centrará en el análisis de documentos opinados obtenidos desde Twitter, lo cual puede ser extendido posteriormente a trabajar con cualquier otro sitio de microblogging que se desee en caso de que esta disponga de un canal de libre acceso a sus documentos.

Los documentos de microblogging tienen la particularidad de ser limitados en la cantidad de caracteres que pueden contener, imponiendo así interesantes desafíos a la hora de analizarlos bajo el alero de técnicas de text mining, permitiendo así gran flexibilidad en los tipos de técnicas y algoritmos que son factibles de utilizar para extraer la información sobre la opinión contenida en estos documentos.

[1] Micro-artículos publicados por los usuarios de *Twitter*

En numerosas publicaciones [5,15,37] se ha tratado el procesamiento de mensajes de *microblogging* o de *Twitter* en diversas aplicaciones, y en todas se concluye que no es posible aplicar un acercamiento tradicional a la extracción de información desde estos debido a su naturaleza ruidosa y dispersa, dificultando la aplicación de las técnicas de procesamiento de lenguaje natural o de minado de texto que se utilizan frecuentemente en text mining.

En el caso particular de Twitter, además de ser documentos de microblogging, estos poseen una gran cantidad de metadata que puede ser utilizada para agrupar los documentos por zona geográfica desde donde fueron publicados, idioma y otras variables disponibles junto con el documento en si, lo que permite realizar una serie de análisis que no serían posibles si sólo se contara con el contenido del documento, tales como obtener la opinión de una zona geográfica sobre un hecho en particular, u obtener sólo opiniones escritas en un lenguaje en particular.

De ahora en adelante, para referirse a un documento opinado extraído desde Twitter, se utilizará el concepto introducido en el capítulo 2, *tweet*, que es el nombre popularmente utilizado para referirse a documentos de microblogging que han sido publicados en esta plataforma.

5.1.1. APIs de Twitter

Para extraer documentos opinados desde Twitter, sus desarrolladores han puesto a disposición de sus usuarios múltiples APIs[2] las cuales permiten recuperar tweets bajo los criterios que uno defina dentro del universo de aquellos que no han sido marcados como privados por el autor, independiente de donde haya sido generado y el idioma en el cual fue escrito. De estas las dos más importantes, la API de *Streaming* y la API de búsqueda, serán descritas a continuación para dar a conocer sus potencialidades y las razones de escoger una por sobre la otra en la implementación del módulo de minado de opiniones.

API de Streaming

El API de Streaming de Twitter posee un enfoque *push*, es decir, el usuario de la API sólo se dedica a esperar notificaciones por parte del servicio en base a los parámetros que fueron escogidos al conectarse a este último. En otras palabras, una API con enfoque push *empuja* la información hacia el usuario, y si la API así lo permite, sólo da a conocer a este aquellas notificaciones que son relevantes de acuerdo a criterios de búsqueda fijados al momento de establecer la conexión entre ambas partes.

Desde el punto de vista general, una API de tipo push tiene como objetivo principal permitir

[2] **API - Application Programming Interface**: Es una especificación que busca ser una interface para comunicar distintos software o componentes dentro de un ecosistema común de aplicaciones.

que el usuario reciba al momento mismo de su creación, nuevos documentos asociados los términos de búsqueda a los cuales se ha suscrito.

Dado lo anterior, una de las limitaciones de las APIs push es que una vez que el evento ya ha ocurrido no se enviarán notificaciones retroactivamente a usuarios que se conecten en un instante de tiempo posterior a aquel donde sucedió el evento. Debido a esto, este tipo de APIs no son utilizables en todo tipo de aplicaciones ni ambientes de desarrollo, un ejemplo claro de esto es cuando la existencia de problemas de conexión entre el cliente y el servidor es frecuente, lo que provocaría perder mucha información cada vez que la comunicación entre ambas partes se vea interrumpida.

Además, la limitación anterior tiene como consecuencia que esta API no pueda ser utilizada cuando se requiere recuperar información del pasado y no sólo desde el momento en que se definieron los términos a buscar, lo que reduce la utilidad de esta para la detección de tendencias en la web, ya que una vez detectados los tópicos pertenecientes a un periodo, es necesario recopilar las opiniones que los usuarios de las redes sociales tienen sobre este en el mismo periodo que ya ha pasado.

API de búsqueda

El API de búsqueda de Twitter complementa a la API de Streaming, ya que posee un enfoque *pull*, el cual consiste en que el usuario pide la información que busca y no está siendo notificando constantemente sobre nuevos documentos que sean generados por los usuarios del sitio.

Una API pull, a diferencia de una API push, no tiene como objetivo transmitir documentos nuevos a medida que estos están siendo creados, si no que otorgar al usuario la capacidad de recuperar documentos ya existentes en la plataforma según los criterios de búsqueda que este defina, por lo que si se desea recibir alertas sobre la aparición de ciertos términos en la plataforma, utilizar este tipo de API es infactible debido a la gran cantidad de consultas que deberían hacerse de manera continua para igualar el bajo tiempo de respuesta de una API de push.

Para que los documentos opinados obtenidos desde Twitter puedan ser utilizados por la plataforma de detección de tendencias, estos deben pertenecer al mismo periodo en el cual se recuperaron los documentos que pertenecen al tópico, por lo que es necesario tener acceso a tweets previamente publicados.

Considerando todos los factores mencionados, y las ventajas y desventajas de cada API, la que más se adecua para ser implementada en la plataforma de detección de tendencias es la de búsqueda, la que será utilizada para recuperar tweets asociados a cada tópico durante cada periodo que esté bajo análisis.

Sin embargo, una limitación importante que se debe considerar a la hora de trabajar con esta API es que su uso está limitado a una cantidad fija $\bar{T}$ de consultas por hora, por lo que se debe

diseñar un algoritmo de recuperación de documentos que sea capaz de recuperar todos los tweets asociados a cada tópico dentro del periodo actual sin sobrepasar el límite de consultas por hora que impone Twitter.

5.2. Recuperación de documentos opinados

En el capítulo 3 se dio a conocer todo flujo de trabajo de la plataforma de detección de tendencias, en el cual se observa que una vez que el módulo de minado de tópicos haya recuperado los documentos presentes en las fuentes sobre las que se está trabajando y haya determinado qué tópicos representan el corpus del periodo actual, es cuando este módulo de la plataforma entra en acción para obtener el sentimiento asociado a cada tópico.

Tal como se mencionó en la introducción del presente capítulo, el asociar a cada tópico un score de sentimiento requiere varias etapas, la primera de ellas es el recuperar todo documento opinado que esté asociado al tópico en cuestión para utilizar algoritmos de minado de opiniones sobre estos y así calcular el score de opiniones en el periodo que está bajo análisis.

Para recuperar documentos opinados se hará uso de el algoritmo presentado en 5.2.1, el cual tiene como datos de entrada una estructura de tópicos $\mathcal{T}$, compuesta por un conjunto de tópicos τ cada uno de los cuales es caracterizado por el conjunto $\mathcal{W}^\tau$ que contiene cada palabra w que lo describe, junto con su distribución de probabilidades ϕ^τ con cada ϕ^τ_w la probabilidad de que el tópico sea τ dado que la palabra es w o $\Pr(\tau \mid w)$; y un periodo de tiempo t caracterizado por una fecha de inicio t^i y una fecha de término t^f.

Algoritmo 5.2.1: Recuperación de documentos opinados

Data: $\mathcal{T}, t$

Result: $\{d_i\}_{i \in \mathbb{N}}$

```
queries = [];
forall τ ∈ 𝒯 do
    queries[τ] = generateQueries(τ);
documents = [];
forall document in Service.query(queries, t^i, t^f) do
    documents = documents ⋃ {document};
return documents;
```

Una vez recibidos los datos de entrada necesarios para su funcionamiento, este algoritmo procede generar las consultas necesarias para extrar los documentos que tengan alguna relación con los tópicos

en cuestión y luego, haciendo uso de éstas, utiliza la interfaz provista para el servicio que se debe minar (en el caso de este trabajo el servicio es la API de Twitter) para todas y cada una de las consultas generadas, recuperando así un conjunto de documentos opinados que una vez almacenados en la base de datos, se guardará también la relación entre si y el tópico en base al cual ha sido recuperado.

5.2.1. Creación de consultas

A continuación se dará a conocer la metodología que se utilizará en este trabajo de investigación para crear el conjunto de conjuntas que será utilizado para recuperar tweets asociados a un tópico τ en un periodo t. Al igual el algoritmo de recuperación de documentos, los datos de entrada del algoritmo generador de consultas tiene como datos de entrada el tópico τ sobre el cual se quiere consultar a Twitter, que es caracterizado por el conjunto de palabras $\mathcal{W}^\tau$ que describen al tópico τ con su distribución de probabilidades ϕ^τ y un periodo t con fecha de inicio t^i y fecha de término t^f.

El algoritmo presentado en 5.2.2 tiene dos constantes fijas que no son utilizadas como datos de entrada si no como variables del modelo en si, que son la cantidad de palabras N que describen al tópico que ser utilizarán para crear las consultas, y la cantidad de palabras n que compondrán a cada consulta que se genere.

Algoritmo 5.2.2: Creación de consultas para recuperación de documentos opinados

---

Data: $\tau = \{\mathcal{W}^\tau, \phi^\tau\}, n, N$

Result: $\mathcal{Q} = \{Q_i\}_{i=1\ldots\binom{N}{n}}$

1 $\mathcal{W}^\tau = \texttt{subset}(\texttt{sort}(\mathcal{W}'^\tau, \phi^\tau), N)$ ordenar $\mathcal{W}^\tau$ en base a ϕ^τ y obtener los N mayores;

2 $\mathcal{Q} = \texttt{subsets}(\mathcal{W}^\tau, n)$ obtiene todos los subconjuntos de largo n;

3 **return** $\mathcal{Q}$;

Inicialmente, el algoritmo se encarga de obtener las N palabras con mayor probabilidad de pertenecer al tópico τ, es decir las N palabras con mayor valor ϕ^τ_w. Luego, tomando el conjunto de palabras $\{w_i\}_{i=1\ldots N}$ se generan de manera aleatoria todos los $n - gramas$ posibles, los cuales serán utilizados posteriormente para realizar consultas con la API de Twitter.

5.2.2. Uso de la API de Twitter para la recuperación documentos opinados

De acuerdo a lo discutido en la sección anterior, la API de Twitter a utilizar será la de búsqueda, la que recibe como parámetros consultas (strings compuestos de una o más palabras) y devuelve como resultado el conjunto de *tweets* asociados a esa búsqueda con hasta un máximo de 10 días de antigüedad.

Dado un superconjunto $\mathcal{Q}_t$ compuesto de todos los conjuntos de consultas $\mathcal{Q}^\tau$ para cada todo tópico τ que haya sido detectado en el periodo t generadas con el algoritmo anterior, y un límite $\bar{\mathcal{Q}}$ de consultas por hora se puede definir un algoritmo de recuperación de tweets como sigue:

Algoritmo 5.2.3: Recuperación de documentos opinados desde Twitter

Data: $\mathcal{Q}_t, \bar{\mathcal{Q}}$

Result: $\{tweet_i\}_{i \in \mathbb{N}}$

```
1 while Q_t ≠ ∅ do
2     queries = pop(Q_t, Q̄) obtiene Q̄ consultas desde Q_t removiéndolas del conjunto;
3     forall query in queries do
4         tweets = tweets ⋃ twitterAPI.search(query, t^i, t^f);
5     waitUntilNextHour();
```

Luego de recuperar cada documento opinado este debe ser guardado en la base de datos junto con toda la metadata asociada a éste, en particular, se almacenará en caso de que sea provisto por la API, la ubicación geográfica (latitud y longitud desde donde se publicó), el idioma en el que está escrito el tweet y la fecha y hora de publicación. Además es necesario almacenar la relación entre el tópico τ y el documento d para posteriormente utilizar esta relación a la hora de asociar todos los puntajes de opiniones de cada documento con los tópicos a los que estos corresponden.

5.3. Minado de opiniones desde documentos de *microblogging*

Una vez que se han recuperado múltiples documentos opinados para cada tópico τ que se esté analizando, se debe proceder a extraer los puntajes de opinión para cada uno de los tweets correspondientes. Según lo visto en el capítulo 2, hay múltiples enfoques a la hora de extraer el sentimiento consignado en un documento opinado, sin embargo, para efectos de este trabajo de investigación se utilizará un algoritmo basado en un lexicón de palabras, el cual, a cada palabra w existente en el lexicón, le asigna puntajes w^p, w^o, w^n para señalar que grado de positividad, neutralidad o negatividad tiene la palabra independiente del contexto en que esta sea utilizada.

Si bien hay una gran variedad de lexicones generados por otros investigadores a lo largo de los últimos años, en este trabajo será utilizado el lexicon de *SentiWordNet* debido a su facilidad de uso y gran cantidad de lexicones clasificados.

Para minar los puntajes de opinión $\vec{o}_d = \left(o_d^p, o_d^n, o_d^n\right)$ desde un documento d, compuesto por un conjunto de palabras $\mathcal{W} = \{w \mid w \in d\}$, se puede utilizar la ecuación 5.1:

$$o_d^i = \frac{\sum_{w \in \mathcal{W}} o_w^i}{\|\mathcal{W}\|} \forall i \in \{o, n, p\} \tag{5.1}$$

5.4. Asociación de opiniones a tópicos

Una vez que cada uno de los documentos opinados recuperados han sido clasificados y se le ha asignado un puntaje de opinión a cada uno de ellos, es necesario asociar esta información a cada uno de los tópicos a los que apuntan estos documentos, logrando así conectar los artículos recuperados desde la web con las opiniones consignadas por los usuarios de las redes sociales.

Utilizando la información recopilada en el proceso de minado de opiniones, se tiene que para cada tópico τ que fue extraído en el periodo t, existe un conjunto de documentos $\mathcal{D}_\tau = \{d \mid Topic(d) = \tau\}$ ya clasificados con su respectivo puntaje de opinión y asociados al tópico τ.

En base a lo anterior, se puede definir el puntaje de opinión para un tópico τ dado su conjunto de documentos $\mathcal{D}_\tau$ a través de la ecuación 5.2:

$$\vec{o}_\tau = (o_\tau^p, o_\tau^n) = \left(\frac{\sum o_d^p}{\|\mathcal{D}_t\|}, \frac{\sum o_d^n}{\|\mathcal{D}_t\|}\right) \tag{5.2}$$

Luego, haciendo uso de la ecuación 5.2, es posible determinar la opinión generalizada de los usuarios de las redes sociales minadas para cada tópico τ perteneciente una estructura de tópicos $\mathcal{T}$ extraída en un periodo t con el siguiente algoritmo:

Algoritmo 5.4.1: Asociación de opiniones a una estructura de tópicos

Data: $\mathcal{D}_t, \mathcal{T}$

1 **forall** $\tau \in \mathcal{T}$ **do**

2 $\quad \tau$.`positiveScore` $= o_\tau^p$;

3 $\quad \tau$.`negativeScore` $= o_\tau^n$;

4 $\quad \tau$.`objectiveScore` $= 1 - (o_\tau^p + o_\tau^n)$;

5.5. Modelo de Datos

Al igual que en el capítulo anterior, en esta sección se presenta el modelo de datos que se utilizará para almacenar los tweets recuperados junto con sus puntajes y relaciones con cada uno de los tópicos sobre los que se trabajo.

En esta sección se presenta una propuesta de modelo de datos para almacenar toda la información recopilada a lo largo de todo el proceso de recuperación de documentos y extracción de tópicos. Este modelo de datos será el encargado de almacenar las noticias, los tópicos y las relaciones entre estos.

Siguiendo con lo sugerido en las secciones anteriores, para cada tweet se deben almacenar: la hora y fecha de publicación, el idioma, la *url* del tweet en la aplicación web de Twitter, el identificador único que se le ha asignado, el contenido sin procesar, la fecha en la cual fue añadida a la base de datos y toda la metadata asociada al documento, la cual al igual que en el caso de los documentos recuperados para el minado de tópicos, se almacenará de manera serializada.

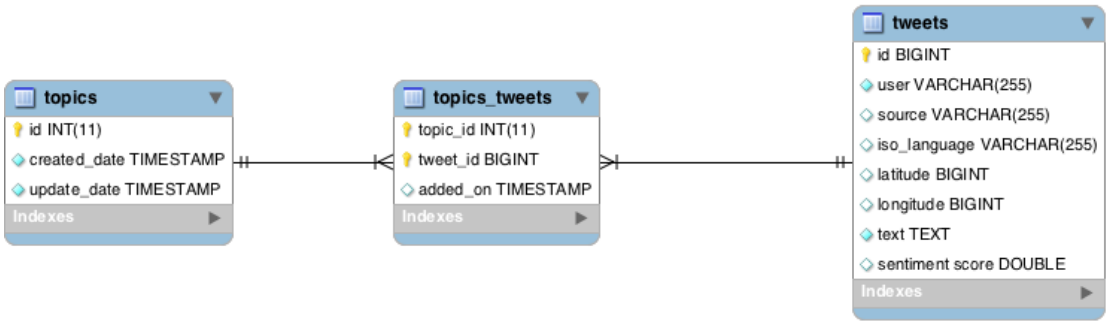

Figura 5.1: Modelo de datos para la recuperación de noticias y la extracción de opiniones

Y por último, es necesario almacenar todos los puntajes de opinión obtenidos para cada uno de los tópicos almacenados y las relaciones entre estos y los tweets que se utilizaron para obtenerlo.

Capítulo 6

Experimentos

La metodología presentada en esta tesis fue llevada a la práctica sobre un conjunto de 20 fuentes que discuten tópicos de tecnología por un periodo de 8 meses. Se hizo uso de fuentes *RSS* debido a que de todas las fuentes de contenido es aquella que más *metadata* expone, y también dado que la manera en que los documentos son expuestos a través de estas fuentes es fácil de procesar y permite hacer peticiones de manera pasiva en busca de nuevos documentos sin la necesidad de abusar de los servidores de los proveedores de contenido.

Por cada documento recuperado desde las fuentes *RSS*, la siguiente información fue almacenada: contenido original en formato *HTML*, fecha de publicación, url original, fuente que lo publicó, fecha y hora de creación y metadata extra presente en el *RSS*.

6.1. Recolección y procesamiento de datos

Cada documento recuperado por el proceso de crawling es almacenado tal como la información proviene de la fuente, es decir, con tags *HTML*, links externos, links de navegación, etc.. Previamente a que sea utilizado tanto por el modelo de tópicos como por el modelo de opiniones estos son procesados a través de la aplicación de técnicas de limpieza de datos tales como la remoción de elementos *HTML*, extracción de *stop-words* y *stemming*. El crawler utilizado para recuperar documentos de blogs o sitios de noticias posee la capacidad de actualizar los documentos si estos cambian luego de que fueran almacenados inicialmente siempre y cuando estos cambios hayan sido registrados de manera explícita por la fuente desde la cual se minó.

6.2. Minado de tópicos

6.2.1. Diseño del experimento

El siguiente experimento tiene como objetivo evaluar el modelo de minado de tópicos propuesto. En particular, se desea evaluar la capacidad de este para detectar el desarrollo de tópicos a lo largo

del tiempo.

Para este experimento, se considerará que un tópico τ se desarrolla del periodo t al periodo $t+1$ si al menos un tópico τ' cumple los criterios mencionados en el capítulo 4.

6.2.2. Criterio de Evaluación

Para evaluar este experimento, para cada tópico τ detectado en el periodo t, se evaluará manualmente si el tópico fue desarrollado en el periodo $t+1$. Sean:

1. TDC = Cantidad de tópicos cuyo desarrollo fue detectado correctamente.
2. TDA = Cantidad de tópicos que tuvieron desarrollo entre t y $t+1$ según el algoritmo.
3. TDM = Cantidad de tópicos que se desarrollaron entre t y $t+1$ encontrados manualmente.

La precisión del modelo de tópicos es calculada a través de la ecuación 6.1, su recall por la ecuación 6.2, y su F-measure por la ecuación 6.3.

$$Precision_{TD} = \frac{TDC}{TDA} \tag{6.1}$$

$$Recall_{TD} = \frac{TDC}{TDM} \tag{6.2}$$

$$FMeasure_{TD} = 2 \cdot \frac{Precision_{TD} \cdot Recall_{TD}}{Precision_{TD} + Recall_{TD}} \tag{6.3}$$

6.2.3. Resultados y Discusión

Para este experimento, se obtuvo una $Precision_{TD}$ de 0.56 y un $Recall_{TD}$ de 0.52, lo que da un $FMeasure_{TD}$ de 0.54. Estos resultados están por sobre el realizar una asignacíon aleatoria de los tópicos, por lo que se puede considerar aceptable el utilizar este algoritmo para realizar esta prueba de concepto debido a su simplicidad y buen rendimiento.

6.3. Minado de opiniones

6.3.1. Diseño del experimento

El segundo experimento realizado busca evaluar los resultados del modelo de opiniones propuesto, el cual se divide en dos pasos: generar consultas para extraer documentos opinionados desde Twitter, y aplicar técnicas clásicas de minado de opiniones sobre los documentos extraídos. Se evaluará primero la generación de consultas y luego el minado de opiniones a partir de los documentos recuperados por ellas.

6.3.2. Criterio de Evaluación

Para la generación de consultas, para cada tópico encontrado en cada periodo t se generan las consultas y luego se extraen documentos de Twitter utilizado estas queries. Una vez estos documentos sean recuperados, se analizará si ellos se refieren o no al tópico en cuestión, así se tiene:

1. $QGDTC$ = Cantidad de documentos que se refieren al tópico correspondiente.
2. $QGRD$ = Cantidad de documentos recuperados por el algoritmo.

Con lo que la precision de la generación de queries viene dada por:

$$Precision_{QG} = \frac{QGDTC}{QGRD} \tag{6.4}$$

Luego, para analizar la segunda parte del algoritmo, considerando las siguientes definiciones:

1. TCP_p = Tweets clasificados correctamente con polaridad p por el algoritmo.
2. TCM_p = Tweets clasificados manualmente con polaridad p.
3. TCA_p = Tweets clasificados por el algoritmo con polaridad p.

Se definen las métricas de *precision* y *recall*:

$$precision\ at\ p = \frac{TCP_p}{TCA_p} \tag{6.5}$$

$$recall\ at\ p = \frac{TCP_p}{TCM_p} \tag{6.6}$$

6.3.3. Resultados y Discusión

Para la generación de queries se obtuvo una $Precision_GQ$ de 0.38, sin embargo, muchos de los documentos recuperados son documentos cuya polaridad no afecta el resultado del análisis de opiniones. Por esto, se considera que una precisión de 0.38 es razonable a la hora de obtener un resultado aproximado de cómo es visto un tópico en las redes sociales desde un punto de vista de opinión.

En la tabla 6.1 se presentan los resultados del experimento relacionado con el minado de opiniones, para cada polaridad.

Estos valores muestran que el algoritmo de minado de opiniones se comporta correctamente. Aún cuando los resultados pueden parecer bajos, debido a que el problema de minado de opiniones de

Polaridad	P	R
Positiva	0.6	0.59
Objetiva	0.53	0.49
Negativa	0.61	0.58

Cuadro 6.1: Precision y recall por polaridad del algoritmo de minado de opiniones

difícil por la cantidad de variables que entran en juego el precision y el recall están dentro de los rangos aceptables para este tipo de algoritmos.

6.4. Modelo de Tendencias

6.4.1. Diseño del experimento

El propósito de evaluar este modelo es determinar su capacidad de representar como los medios reaccionan frente a eventos que ocurren en un periodo que se esté realizando el análisis. Los eventos sobre los que se centra este experimento serán llamados *eventos significativos* y son definidos como:

Definición 6.1. Evento Significativo: Si entre dos periodos consecutivos de tiempo t_i y t_{i+1}, la diferencia entre la cantidad de documentos publicados por los medios $\frac{\|\vec{D}_{t_{i+1}}\|-\|\vec{D}_{t_i}\|}{\|\vec{D}_{t_i}\|}$ es mayor que un umbral ρ, entonces se dice que ocurrió un evento significativo en t_{i+1}.

Un ejemplo de un tópico que contiene un evento significativo puede ser visto en 6.1, en la cual el sentimiento asociado a él es mostrado como una curva y la cantidad de documentos de noticias son representadas como barras. Debido al gran incremento en la cantidad de estos últimos entre los periodos 5 y 6, se señala un evento significativo en el periodo 6. Por otro lado, también es posible observar como la cobertura por los medios y el sentimiento en las redes sociales sobre este tópico cambian a lo largo del tiempo.

6.4.2. Criterio de Evaluación

Para evaluar el framework en su totalidad, se tomó el siguiente enfoque: para cada tópico, la serie de tiempo que le corresponde se analizará de manera manual en busca de *eventos significativos*, y la precisión del framework será calculada en base a su comportamiento en relación a eventos significativos, es decir, si un evento importante sucedió en el mismo periodo en el que la metodología detecto un evento significativo, entonces se cuenta como un éxito. Sean:

1. $SECC$ = Eventos significativos correctamente clasificados como tales (proceso manual).
2. $ASEM$ = Cantidad de eventos significativos encontrados por el algoritmo.
3. $ASEF$ = Cantidad de eventos significativos encontrados manualmente.

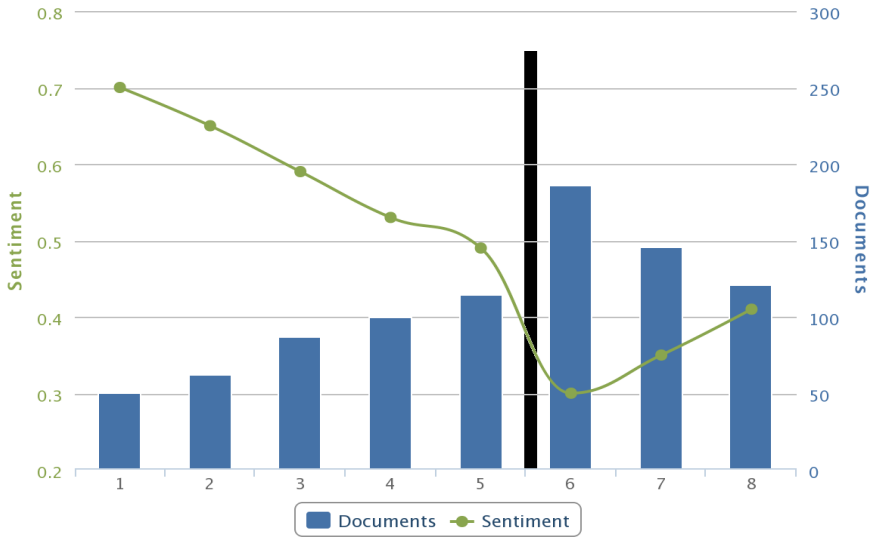

Figura 6.1: Evolución de un tópic a lo largo del tiempo

Eventos que puedan ser considerados "eventos significativos" son clasificados de manera manual. Por otro lado, $ASEM$ es calculado haciendo uso del algoritmo presentado en 6.4.1.

Algoritmo 6.4.1: Calculo de eventos significativos

Data: $Topics$

forall $\tau \in Topics$ **do**
  forall $t \in \tau$ **do**
    if $\frac{\|\vec{D}_{t_{i+1}}\| - \|\vec{D}_{t_i}\|}{\|\vec{D}_{t_i}\|} > \rho$ **then**
      $EV_\tau + +;$

La precisión del framework es calculada a través de la ecuación 6.7, su recall por la ecuación 6.8, y su F-measure por la ecuación 6.9.

$$Precision_{SE} = \frac{SECC}{ASEM} \tag{6.7}$$

$$Recall_{SE} = \frac{SECC}{ASEF} \tag{6.8}$$

$$FMeasure_{SE} = 2 \cdot \frac{Precision_{SE} \cdot Recall_{SE}}{Precision_{SE} + Recall_{SE}} \tag{6.9}$$

6.4.3. Resultados y Discusión

Luego de que el experimento propuesto fue realizado, se recuperaron un total de 200.890 documentos, de los cuales 117 tópicos fueron extraídos, y finalmente se recuperaron 268.800 tweets.

Además, un total de 65 *eventos significativos* distribuidos a lo largo de todos estos tópicos fueron detectados de manera manual.

ρ	$Precision_{SE}$	$Recall_{SE}$	$FMeasure_{SE}$
0.2	0.25	0.71	0.37
0.3	0.38	0.65	0.48
0.4	0.48	0.57	0.52
0.5	0.61	0.51	0.56

Cuadro 6.2: $Precision_{SE}$, $Recall_{SE}$, y $FMeasure_{SE}$ para múltiples valores de ρ

Para valores de ρ mayor o iguales a 0.6, no se encontraron eventos significativos dado que la cantidad de noticias en un periodo semanal para tópicos sobre tecnología no pueden alcanzar la cantidad de noticias nuevas necesarias para que un evento pueda ser clasificado como significativo.

Capítulo 7

Conclusiones y Trabajo Futuro

En este trabajo se exploró el potencial de las aplicaciones de detección de tendencias con el objetivo de modelar la demanda de productos y servicios a través de la información presente en la Web. La creciente necesidad de las empresas por poseer más información a la hora de tomar decisiones, y hacer uso de esta información en el menor tiempo posible, abre la oportunidad de desarrollar este tipo de aplicaciones.

Debido a que hoy en día la mayor fuerte de información es la Web, la aplicación desarrollada hace uso tanto de documentos de noticias como opinados para crear una representación comparable de los tópicos a lo largo del tiempo. Así, permite el desarrollo sobre si mismo de nuevas metodologías y extensiones enfocadas en la detección y el modelamiento de las tendencias en la Web incluso en otros dominios del conocimiento, no solo el utilizado a lo largo de esta investigación.

Aún cuando existen modelos de tópicos más robustos que el aquí presentado, se optó por realizar una solución de bajo costo computacional con el fin de poder procesar más documentos y por ende obtener más información sobre los hechos que están siendo cubiertos en la Web. Esto va de la mano con mantener el foco en realizar una prueba de concepto y demostrar el potencial existente en la detección de tendencias.

En relación a los objetivos propuestos para este trabajo, todos fueron cumplidos exitosamente. En el capítulo 2 se estableció el estado del arte de todas las ramas de investigación que son utilizadas en este trabajo, junto con dar a conocer aplicaciones de estos que fueran útiles para el trabajo desarrollado. Posteriormente, se diseñó e implementó algoritmos de recolección de documentos, minado de tópicos a lo largo del tiempo y Web Opinion Mining. Finalmente, se desarrolló una visualización de tendencias que permitiera identificar el impacto de un tópico a lo largo del tiempo.

Además, luego de realizar los experimentos presentados en el capítulo anterior, se concluye que la metodología presentada en este trabajo es un enfoque factible para modelar tendencias en la Web a través de la interacción de eventos, tópicos y opiniones consignadas en la Web 2.0. Así, también

se concluye que el trabajo desarrollado cumple con el objetivo de realizar una prueba de concepto de detección de tendencias, y asi mismo con el objetivo general de este trabajo.

Por otro lado, los experimentos realizados comprueban la hipótesis planteada al inicio de este trabajo, ya que una vez toda la información es recolectada y analizada, es posible analizar el comportamiento de los tópicos a lo largo del tiempo, y ver como reaccionan los usuarios de la Web 2.0 y de manera indirecta caracterizar la demanda sobre ciertos productos y servicios.

En base a lo expuesto anteriormente, la contribución de este trabajo es mostrar la factibilidad de de realizar un modelo de detección de tendencias en base a noticias y opiniones consignadas en la Web 2.0. Además, provee los lineamientos necesarios para mejorar el modelo propuesto y expandir sus casos de uso.

Finalmente, debido a la amplia definición de lo que es una tendencia, y el aún más amplio espectro de variables que pueden ser tomadas en consideración para modelarlas, debe tenerse en mente que esta investigación propone una metodología inicial para este tipo de aplicaciones. Por lo tanto, este trabajo fue desarrollado para que fuese extensible.

7.1. Trabajo futuro

Debido a que esta prueba de concepto que logra mostrar el potencial de este tipo de aplicaciones y la gran variedad de áreas que están involucradas en este trabajo de investigación, hay muchas oportunidades para extender este trabajo. El trabajo propuesto se dividirá en tres secciones, aquellas extensiones teóricas que pueden apuntar a mejorar cada uno de los modelos utilizados en la detección desde un punto de vista teórico; aquellas extensiones prácticas que apunten a mejorar la información propuesta o la manera en que esta es manipulada; y finalmente, aquellos proyectos, funcionalidades o mejoras que complementen el trabajo de detección de tendencias con el fin de obtener una herramienta de monitoreo de la web.

7.1.1. Extensiones teóricas

En esta categoría caen todas aquellas extensiones que apunten a mejorar la plataforma en el área teórica en las áreas de recuperación de la información y de minado de datos. A continuación de detalla una lista de posibles extensiones del modelo actual:

1. Probar modelos de extracción de tópicos más elaborados.

2. Modificar el algoritmo de recuperación de documentos que alimenta el módulo de minado de tópicos para que sea capaz de agregar nuevas fuentes de manera no supervisada, o bien, que

este sea capaz de sugerir fuentes que potencialmente sean un aporte positivo al universo donde está procesando información el modelo actual.

3. Mejorar el modelo de detección de opiniones para que considere de manera intrínseca la evolución de tópicos en el tiempo.

4. Mejorar el modelo de extracción de opiniones para que considere léxicos propios de un ecosistema de microblogging tales como emoticones (por ej. `:)`, `:(` y acrónimos (por ej. *lol, omg*).

5. Permitir que el módulo de minado de opiniones extraiga opiniones desde otro tipo de documentos, no sólo aquellos de microblogging.

6. Proponer algoritmos y heurísticas que permitan la detección de tendencias de manera no supervisada.

7.1.2. Extensiones prácticas

Todos aquellos desarrollos futuros que apunten a crear nuevas funcionalidades o enfocar esta herramienta a otros usos son considerados dentro de esta categoría.

1. Creación de una herramienta de monitoreo de valor de marca en la web.

2. Medición de impacto de campañas publicitarias.

3. Predicción de variaciones en el precio de portafolios de acciones.

7.1.3. Monitoreo de la Web

Se consideran en esta categoría todas aquellas aplicaciones que busquen complementar la plataforma de detección de tendencias con miras a la creación de una plataforma de monitoreo de la web. Ejemplo de herramientas complementarias a lo desarrollado en este trabajo son:

1. Desarrollo de un sistema de alerta temprana el cual, a través del monitoreo de las redes sociales y de las noticias sea capaz de detectar acontecimientos importantes apenas estos sucedan.

2. Desarrollo de un sistema de monitoreo de tópicos de manera permanente. Esta plataforma, a diferencia de la de detección de tendencias, se enfocará en analizar el comportamiento de los usuarios de la web y de las fuentes de publicación de contenido en torno a un tópico en particular.

7.1.4. Roadmap

A continuación se propone un *roadmap* para el desarrollo del trabajo futuro en base a lo presentado. Inicialmente, se sugiere explorar el rendimiento del modelo con otro tipo de modelo de tópicos. Luego, mejorar el algoritmo de minado de opiniones para considerar las distintas características inherentes a ellas.

Una vez que ambas posibilidades han sido exploradas, se recomienda desarrollar modelos no supervisados de detección de tendencias. Este tipo de investigación se ve limitada debido a lo amplia que puede llegar a ser la definición de tendencia, y por otro lado la dificultad de encontrar datos o información con la que comparar los modelos propuestos. Además, dependiendo de la definición de tendencia que se utilice, es posible realizar supuestos estadísticos muy fuertes que también amenacen la validez de la hipótesis propuesta.

Una vez que las mejoras teóricas han sido realizadas, se recomienda explorar la utilidad de este tipo de modelos en otros nichos y no sólo en el análisis de demanda por productos y servicios de mercados particulares.

Referencias

[1] F. Alvanaki, M. Sebastian, K. Ramamritham, and G. Weikum. Enblogue: emergent topic detection in web 2.0 streams. In *Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data*, SIGMOD '11, pages 1271–1274, Athens, Greece, 2011. ACM.

[2] N. Archak, A. Ghose, and P. G. Ipeirotis. Show me the money!: deriving the pricing power of product features by mining consumer reviews. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '07, pages 56–65, San Jose, California, USA, 2007. ACM.

[3] S. Baccianella, A. Esuli, and F. Sebastiani. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In N. C. C. Chair), K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner, and D. Tapias, editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation*, LREC '10, pages 2200–2204, Valletta, Malta, may 2010. European Language Resources Association (ELRA).

[4] R. Baeza-Yates, B. Ribeiro-Neto, et al. *Modern information retrieval*, volume 463. ACM press New York., 1999.

[5] A. Bermingham and A. F. Smeaton. Classifying sentiment in microblogs: is brevity an advantage? In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 1833–1836, Toronto, Ontario, Canada, 2010. ACM.

[6] T. Berners-Lee. Information Management: A Proposal, March 1989.

[7] M. P. Bill Heil. New twitter research: Men follow men and nobody tweets. `http://blogs.hbr.org/cs/2009/06/new_twitter_research_men_follo.html`, 2009.

[8] D. M. Blei and J. D. Lafferty. Correlated topic models. In *NIPS*, 2005.

[9] D. M. Blei and J. D. Lafferty. Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, ICML '06, pages 113–120, Pittsburgh, Pennsylvania, 2006. ACM.

[10] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, Mar. 2003.

[11] R. Blood. Weblogs: A history and perspective, 2000.

[12] F. Bodendorf and C. Kaiser. Detecting opinion leaders and trends in online communities. In *Proceedings of the 2010 Fourth International Conference on Digital Society*, ICDS '10, pages 124–129, Washington, DC, USA, 2010. IEEE Computer Society.

[13] D. M. Boyd and N. B. Ellison. Social network sites: Definition, history, and scholarship. *Journal of Computer-Mediated Communication*, 13(1):article 11, October 2007.

[14] S. Brody and N. Diakopoulos. Cooooooooooooooollllllllllllll!!!!!!!!!!!!!! using word lengthening to detect sentiment in microblogs. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 562–570, Edinburgh, Scotland, UK., July 2011. Association for Computational Linguistics.

[15] S. Carter, M. Tsagkias, and W. Weerkamp. Semi-supervised priors for microblog language identification, February 2011.

[16] M. Craig, O. Iadh, and S. Ian. Overview of the trec 2007 blog track. In *Proceeddings of the 16th Text REtrieval Conference*, TREC '07, 2007.

[17] I. P. Cvijikj and F. Michahelles. Monitoring trends on facebook. In *Proceedings of the 2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing*, DASC '11, pages 895–902, Sydney, Australia, 2011. IEEE Computer Society.

[18] T. Damer. *Attacking faulty reasoning: a practical guide to fallacy-free arguments*. Wadsworth/Cengage Laerning, Australia Belmont, CA, 2009.

[19] S. David and T. J. Pinch. Six degrees of reputation: The use and abuse of online review and recommendation systems. *First Monday*, July 2006. Special Issue on Commercial Applications of the Internet.

[20] P. Domingos. Mining social networks for viral marketing. *IEEE Intelligent Systems*, 20(1):80–82, 2005.

[21] T. Engineering. Delivering 350 billion tweets a day is a terribly fun engineering challenge. `https://twitter.com/#!/twitterglobalpr/status/91892897888206848`, July 2011. Visitada el 15 de Agosto del 2012.

[22] A. Esuli and F. Sebastiani. SentiWordNet: A publicly available lexical resource for opinion mining. In *Proceedings of the Third International Conference on Language Resources and Evaluation*, LREC '06, pages 417–422, Genoa, Italy, 2006. European Language Resources Association (ELRA).

[23] A. Funk, Y. Li, H. Saggion, K. Bontcheva, and C. Leibold. Opinion analysis for business intelligence applications. In *Proceedings of the first international workshop on Ontology-supported business intelligence*, OBI '08, pages 3:1–3:9, Karlsruhe, Germany, 2008. ACM.

[24] A. Ghose, P. Ipeirotis, and A. Sundararajan. Opinion mining using econometrics: A case study on reputation systems. In *Proceedings of the Association for Computational Linguistics*, pages 416–423, Prague, Czech Republic, 2007. Association for Computational Linguistics.

[25] B. Gu, P. Konana, A. Liu, B. Rajagopalan, and J. Ghosh. Predictive value of stock message board sentiments. *McCombs Research Paper No. IROM-11-06*, 2006.

[26] V. Hatzivassiloglou and J. Wiebe. Effects of adjective orientation and gradability on sentence subjectivity. In *Proceedings of the 18th conference on Computational linguistics - Volume 1*, volume 1 of *COLING '00*, pages 299–305. Association for Computational Linguistics, 2000.

[27] M. Hu and B. Liu. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '04, pages 168–177, Seattle, Washington, USA, 2004. ACM.

[28] M. Hu and B. Liu. Mining opinion features in customer reviews. In *Proceedings of the 19th national conference on Artifical intelligence*, AAAI '04, pages 755–760, San Jose, California, USA, 2004. AAAI Press.

[29] L. Huang et al. A survey on web information retrieval technologies. *Computer Science Departament, State University of New York*, 2000.

[30] X. Jin, Y. Li, T. Mah, and J. Tong. Sensitive webpage classification for content advertising. In *Proceedings of the 1st International Workshop on Data Mining and Audience Intelligence for Advertising*, ADKDD '07, pages 28–33, San Jose, California, USA, 2007. ACM.

[31] N. Jindal and B. Liu. Opinion spam and analysis. In *Proceedings of the International Conference on Web Search and Web Data Mining*, WSDM '08, pages 219–230, Palo Alto, California, USA, 2008. ACM.

[32] A. Joshi, A. R. Balamurali, P. Bhattacharyya, and R. Mohanty. C-feel-it: a sentiment analyzer for micro-blogs. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Systems Demonstrations*, HLT '11, pages 127–132, Portland, Oregon, USA, 2011. Association for Computational Linguistics.

[33] H. Kanayama and T. Nasukawa. Textual demand analysis: detection of users' wants and needs from opinions. In *Proceedings of the 22nd International Conference on Computational Linguistics - Volume 1*, COLING '08, pages 409–416, Manchester, United Kingdom, 2008. Association for Computational Linguistics.

[34] A. M. Kaplan and M. Haenlein. The early bird catches the news: Nine things you should know about micro-blogging. *Business Horizons*, 54(2):105–113, 2010.

[35] A. Kennedy and D. Inkpen. Sentiment classification of movie reviews using contextual valence shifters. *Computational Intelligence*, 22:2006, 2006.

[36] E. Kouloumpis, T. Wilson, and J. Moore. Twitter sentiment analysis : The good the bad and the omg ! *Artificial Intelligence*, 70(2):538–541, 2011.

[37] G. Laboreiro, L. Sarmento, J. Teixeira, and E. Oliveira. Tokenizing micro-blogging messages using a text classification approach. In *Proceedings of the fourth workshop on Analytics for noisy unstructured text data*, AND '10, pages 81–88, Toronto, ON, Canada, 2010. ACM.

[38] D. Lee, O.-R. Jeong, and S.-g. Lee. Opinion mining of customer feedback data on the web. In *Proceedings of the 2nd international conference on Ubiquitous information management and communication*, ICUIMC '08, pages 230–235, Suwon, Korea, 2008. ACM.

[39] B. Liu. Opinion mining and summarization - sentiment analysis, 2008.

[40] B. Liu. Sentiment analysis and subjectivity. In N. Indurkhya and F. J. Damerau, editors, *Handbook of Natural Language Processing, Second Edition*. CRC Press, Taylor and Francis Group, Boca Raton, Florida, USA, 2010. ISBN 978-1420085921.

[41] Y. Liu, X. Huang, A. An, and X. Yu. Arsa: a sentiment-aware model for predicting sales performance using blogs. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 607–614, Amsterdam, The Netherlands, 2007. ACM.

[42] M. Mathioudakis and N. Koudas. Twittermonitor: trend detection over the twitter stream. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data*, SIGMOD '10, pages 1155–1158, Indianapolis, Indiana, USA, 2010. ACM.

[43] S. Matsumoto, H. Takamura, and M. Okumura. Sentiment classification using word subsequences and dependency sub-trees. *Advances in Knowledge Discovery and Data Mining*, pages 21–32, 2005.

[44] R. Mihalcea, C. Banea, and J. Wiebe. Learning multilingual subjective language via cross-lingual projections. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 976–983, Prague, Czech Republic, 2007. Association for Computational Linguistics.

[45] G. Mishne. Experiments with mood classification in blog posts. In *Proceedings of the 1st Workshop on Stylistic Analysis Of Text For Information Access (Style 2005)*, SIGIR '05, 2005.

[46] G. Mishne and N. Glance. Predicting movie sales from blogger sentiment. In *In AAAI 2006 Spring Symposium on Computational Approaches to Analysing Weblogs*, CAAW '06, pages 301–304. AAAI Press, 2006.

[47] I. Mogotsi. Christopher d. manning, prabhakar raghavan, and hinrich schütze: Introduction to information retrieval. *Information Retrieval*, 13:192–195, 2010. 10.1007/s10791-009-9115-y.

[48] S. Morinaga, K. Yamanishi, K. Tateishi, and T. Fukushima. Mining product reputations on the web. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, pages 341–349, Edmonton, Alberta, Canada, 2002. ACM.

[49] T. Mullen and N. Collier. Sentiment analysis using support vector machines with diverse information sources. In *Proceedings of Conference on Empirical Methods in Natural Language Processing*, volume 4 of *EMNLP '04*, pages 412–418. ACL, 2004.

[50] S. Murugesan. Understanding web 2.0. *IT professional*, 9(4):34–41, 2007.

[51] V. Ng, S. Dasgupta, and S. Arifin. Examining the role of linguistic knowledge sources in the automatic identification and classification of reviews. In *Proceedings of the COLING/ACL on Main conference poster sessions*, pages 611–618. Association for Computational Linguistics, 2006.

[52] B. O'Connor, R. Balasubramanyan, B. Routledge, and N. Smith. From tweets to polls: Linking text sentiment to public opinion time series. In *Proceedings of the International AAAI Conference on Weblogs and Social Media*, ICWSM '10, pages 122–129. AAAI Press, 2010.

[53] B. Ohana and B. Tierney. Sentiment classification of reviews using sentiwordnet. *Discovery*, page 13, 2009.

[54] G. Paltoglou, S. Gobron, M. Skowron, M. Thelwall, and D. Thalmann. Sentiment analysis of informal textual communication in cyberspace. In *Proceedings of ENGAGE 2010*, ENGAGE '10, pages 13–25, Zermatt, Switzerland, 2010.

[55] B. Pang and L. Lee. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1–135, Jan. 2008.

[56] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing - Volume 10*, EMNLP '02, pages 79–86, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.

[57] G. Pant, P. Srinivasan, and F. Menczer. Crawling the web. In *In Web Dynamics: Adapting to Change in Content, Size, Topology and Use. Edited by M. Levene and A. Poulovassilis*, pages 153–178. Springer-Verlag, 2004.

[58] W. Parrott. *Emotions in Social Psychology: Key Readings*. Key Readings in Social Psychology. Taylor & Francis, 2000.

[59] A.-M. Popescu and O. Etzioni. Extracting product features and opinions from reviews. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 339–346, Vancouver, British Columbia, Canada, 2005. Association for Computational Linguistics.

[60] M. F. Porter. An algorithm for suffix stripping. In K. Sparck Jones and P. Willett, editors, *Readings in information retrieval*, pages 313–316. Morgan Kaufmann Publishers Inc., San Francisco, California, USA, 1997.

[61] J. Read. Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In *Proceedings of the ACL Student Research Workshop*, ACLstudent '05, pages 43–48, Ann Arbor, Michigan, 2005. Association for Computational Linguistics.

[62] L. Sarmento, P. Carvalho, M. Silva, and E. de Oliveira. Automatic creation of a reference corpus for political opinion mining in user-generated content. In *Proceedings of the 1st International CIKM Workshop on Topic-Sentiment Analysis for Mass Opinion*, CIKM '09, pages 29–36, Hong Kong, China, 2009. ACM.

[63] V. Sehgal and C. Song. Sops: stock prediction using web sentiment. In *Data Mining Workshops, 2007. ICDM Workshops 2007. Seventh IEEE International Conference on*, ICDMW '07, pages 21–26, Omaha, Nebraska, USA, 2007. IEEE Computer Society.

[64] M. Silva, P. Carvalho, L. Sarmento, E. de Oliveira, and P. Magalhaes. The design of optimism, an opinion mining system for portuguese politics. *New Trends in Artificial Intelligence: Proceedings of EPIA*, pages 12–15, 2009.

[65] M. Steyvers and T. Griffiths. *Latent Semantic Analysis: A Road to Meaning*, chapter Probabilistic topic models. Laurence Erlbaum, 2007.

[66] J. Tatemura. Virtual reviewers for collaborative exploration of movie reviews. In *Proceedings of the 5th International Conference on Intelligent User Interfaces*, IUI '00, pages 272–275, New Orleans, Louisiana, USA, 2000. ACM.

[67] L. Terveen, W. Hill, B. Amento, D. McDonald, and J. Creter. Phoaks: a system for sharing recommendations. *Commun. ACM*, 40(3):59–62, Mar. 1997.

[68] P. D. Turney. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 417–424, Philadelphia, Pennsylvania, 2002. Association for Computational Linguistics.

[69] B. Walsh. Markov chain monte carlo and gibbs sampling, lecture notes for eeb 581, version 26, April 2004.

[70] T. Wilson, J. Wiebe, and P. Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 347–354, Vancouver, British Columbia, Canada, 2005. Association for Computational Linguistics.

[71] L. Zhang, B. Liu, S. H. Lim, and E. O'Brien-Strain. Extracting and ranking product features in opinion documents. In *Proceedings of the 23rd International Conference on Computatio-*

nal Linguistics: Posters, COLING '10, pages 1462–1470, Beijing, China, 2010. Association for Computational Linguistics.

[72] D. Zhao and M. B. Rosson. How and why people twitter: the role that micro-blogging plays in informal communication at work. In *Proceedings of the ACM 2009 international conference on Supporting group work*, GROUP '09, pages 243–252, Sanibel Island, Florida, USA, 2009. ACM.

Apéndices

A. Publicaciones derivadas de este trabajo

Primero se presenta el trabajo publicado en la *Revista de Sistemas de Ingeniería* el año 2013 bajo el título "*Una aplicación de Web Opinion Mining para la extracción de tendencias y tópicos de relevancia a partir de las opiniones consignadas en blogs y sitios de noticias*".

El segundo trabajo desarrollado es un paper publicado en *Workshop On Social Web Intelligence* que se desarrolla dentro de la conferencia WI 2013. Este trabajo se titula "*Sentiment Polarity of Trends on the Web Using Opinion Mining and Topic Modeling*".

Una aplicación de Web Opinion Mining para la extracción de tendencias y tópicos de relevancia a partir de las opiniones consignadas en blogs y sitios de noticias

Rodrigo Dueñas F.
Juan D. Velásquez*

Resumen

El análisis de tendencias se ha abordado tradicionalmente a través de la realización de encuestas, las cuales poseen un alto contenido de subjetividad y las respuestas se ven constantemente afectadas por factores exógenos al evento bajo estudio. Este exceso de factores exógenos y subjetividad puede conducir a errores significativos, basta con ver los resultados de la última encuesta para la elección de alcaldes, la que predijo de manera errónea qué candidatos ganarían en las comunas mas emblemáticas de Santiago. En este trabajo, presentamos una metodología alternativa para detectar tendencias en la Web, a través del uso de técnicas de recuperación de la información, modelamiento de tópicos y minado de opiniones. Dado un conjunto de sitios semilla, se procede a extraer los tópicos que se mencionan en los documentos recuperados desde ellos y posteriormente se acude a las redes sociales para obtener la opinión por parte de sus usuarios en relación a estos. Usando esta metodología de detección de tendencias es posible complementar la información extraída a través de metodologías tradicionales para predecir eventos y reducir los efectos de los factores exógenos introducidos por los medios tradicionales.

Palabras Clave: *Opiniones*, *Tendencias*, *Web Opinion Mining*, *Tópicos*, *Blogs*, *Noticias*.

*Departamento de Ingeniería Industrial, Facultad de Ciencias Físicas y Matemáticas, Universidad de Chile, Santiago, Chile.

1. Introducción

Con la aparición de las aplicaciones web que permiten la creación de contenido y la colaboración por parte de los usuarios, como lo son *wikis* y *blogs*, (las cuales darían el puntapié inicial a lo que ahora se conoce como la Web 2.0) la función de la Web en la sociedad mundial se vio fuertemente potenciada. Esta dejó de ser tan sólo un repositorio de información y se transformó en un canal interactivo entre todas las entidades que la componen, tanto usuarios como proveedores de información. Este cambio de paradigma permitió que todos ellos pudiesen contribuir activamente a la creación de contenido, provocando un explosivo aumento en la participación de sus usuarios en la Web, y por consiguiente de la cantidad de información y conocimiento disponible en ella.

Junto a ello, Internet trajo consigo cambios drásticos en la manera que se interactúa en el mercado empresarial. Estos cambios provocan que una empresa pueda crecer en muchas direcciones y no sólo aumentando la cantidad de productos que produce o el número de personas a las que ofrece sus servicios. Así, una vez que una empresa decide crecer, ya sea expandiendo negocios hacia nuevos mercados u ofreciendo nuevos productos y servicios, la cantidad de información externa que debe abarcar para poder realizar una buena toma de decisiones se vuelve cada vez mayor, por lo que debe analizar un conjunto siempre creciente de fuentes de información para poder recuperar el conocimiento necesario para que este análisis sea valioso para la empresa.

En esta misma línea, es cada vez más necesario ser capaz de manejar grandes volúmenes de datos para gestionar de la mejor manera posible los recursos que se disponen, y al mismo tiempo anticiparse a cada movimiento que realizará la competencia en busca de obtener ventajas competitivas, o impedir que otros las obtengan, para ser líderes en el mercado. El primer problema al que se debe enfrentar una empresa sumergida en el mundo globalizado, es el más complejo desde el punto de vista de la gestión de operaciones, por lo que varias metodologías y herramientas han nacido proponiendo soluciones, entre las cuales se encuentran los *Data Warehouses*, la *Business Intelligence* y el recientemente acuñado término de *BigData*. El segundo problema no sólo involucra a la gestión de operaciones, ya que es necesario tener un equipo multidisciplinario encargado constantemente de monitorear el mercado, las acciones de las otras empresas, los anuncios presentes en los medios y cualquier indicio que permita anticiparse a los lanzamientos de productos y servicios de la competencia.

Una posible solución al problema planteado es minar la web en busca de esos indicios de manera automática, con foco en qué tópicos se habla en la web, y analizando las redes sociales para estimar que percepción poseen los ciber-

nautas sobre ellos. Es factible plantear la hipótesis de que a través de realizar un análisis de gran parte del conocimiento objetivo generado por los usuarios y los medios se puede atisbar aquellos indicios claves a la hora de plantear una planificación estratégica y operacional. Un sistema capaz de realizar esto de manera aproximada es realizable utilizando técnicas de recuperación de la información, modelamiento de tópicos y minado de opiniones sobre fuentes cuidadosamente seleccionadas que sean capaces de otorgarle al sistema una muestra significativa de todo lo que se habla sobre los mercados en los cuales se ve inmersa la empresa a nivel competitivo.

Por lo mencionado anteriormente, se plantea como hipótesis de investigación que es posible extraer tendencias y obtener una representación aproximada del comportamiento de estas a través del análisis de los documentos presentados en sitios de noticias y las opiniones consignadas en las redes sociales por parte de sus usuarios.

En la segunda sección de este artículo se da a conocer el estado del arte respecto de las técnicas de recuperación de información, modelamiento de tópicos en documentos y finalmente sobre algoritmos de minado de opiniones. En la sección 3, se da a conocer en detalle el modelo propuesto para la detección de tendencias en la Web, en particular la detección de tópicos y el minado de opiniones referentes a estos, los cuales serán evaluados a través de los experimentos presentes en la sección 4. Para finalizar, en la quinta sección de este artículo se presentan las conclusiones relevantes al trabajo desarrollado y posibles ramas de investigación a futuro.

2. Trabajo relacionado

2.1. Modelos de Tópicos

Un modelo de tópicos tiene como objetivo identificar las relaciones latentes entre documentos pertenecientes a una colección, con el fin de dar una descripción sucinta de esta sin perder información desde el punto de vista estadístico.

El precursor de los modelos de tópicos es David Blei, el cual en [4] describe de manera detallada los modelos de tópicos y las aplicaciones de estos. En ella, se define un tópico como el conjunto de elementos que pueden representar una temática presente en una colección de documentos sin pérdida de información estadística. Por ejemplo, si existe una colección de documentos textuales que abarca múltiples temas, un tópico es un conjunto de palabras que logra describir estadísticamente uno de estos temas.

Entre los modelos de tópicos existente, los más utilizados son los desarro-

llados por Blei *et al*. Entre ellos, los más populares son el modelo LDA (Latent Dirichlet Allocation) [4] y el modelo CTM (Correlated Topic Model) [3].

Estos modelos de tópicos se cimentan sobre las siguientes definiciones:

- Una *palabra* w es la unidad elemental de un documento textual y se define como un elemento de un vocabulario indexado V. Para efectos de estos modelos, para representar una palabra se hace uso de vectores unitarios en donde la n-ésima palabra de V se representa con un vector de largo $|V|$ en el cual sólo su componente n-ésima es igual a 1.
- Un *documento* es un arreglo de palabras descrito como $\mathbf{w} = (w_1, w_2, ..., w_N)$, donde w_n es la n-ésima palabra de este.
- Un *corpus* es una colección de documentos descrita como $D = \{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_M\}$.
- Un *tópico* es una distribución de probabilidad sobre un vocabulario V fijo. Por ejemplo, el tópico *política* está descrito por palabras como *partido, diputado, senado, ley* de manera frecuente y palabras como *guerra, marcador, gol* con probabilidad casi nula.

A continuación se da a conocer una descripción de cada uno de los modelos mencionados anteriormente, dando a conocer las diferencias entre estos y las principales características de cada uno de ellos.

2.1.1. Latent Dirichlet Allocation

El modelo llamado *Latent Dirichlet Allocation*[4] es considerado el más sencillo de los modelos de tópicos presentes hoy en día, y por ello es utilizado frecuentemente en aplicaciones que requieran obtener información sobre colecciones de documentos de manera rápida y eficiente.

El modelo LDA trabaja bajo el supuesto de que los tópicos presentes en la colección de documentos que se está analizando no necesariamente están relacionados y por consiguiente no dependen entre ellos.

Para extraer la estructura de tópicos presente en una colección, este modelo hace uso de un modelo estadístico de generación de documentos, tópicos y palabras a lo largo del tiempo que abarque esta. El siguiente proceso se realiza para cada documento presente en una colección:

1. Definir una distribución aleatoria para la presencia de los tópicos en la colección y una distribución para la presencia de las palabras para cada tópico que se desea encontrar.
2. Luego, por cada palabra presente en el documento bajo análisis se debe:
   a) Escoger un tópico aleatoriamente haciendo uso de la distribución generada en el paso 1.

b) Escoger una palabra del documento aleatoriamente a partir de la distribución del vocabulario en relación al tópico escogido.

Formalmente, para determinar la estructura de tópicos existente luego del proceso de generación, es necesario calcular las distribuciones condicionales entre los tópicos y sus documentos, el cual es un problema NP completo debido a que la cantidad de estructuras que pueden representar una colección de documentos crece exponencialmente en relación a la cantidad de documentos y palabras presente en esta. Este proceso es descrito formalmente como sigue:

1. Escoger $N \sim Poisson(\xi)$
2. Escoger $\theta \sim Dirichlet(\alpha)$
3. Para cada palabra w_n en $\mathbf{w}$
   a) Escoger un tópico $z_d \sim Multinomial(\theta)$
   b) Escoger una palabra w_d a partir de $p(w_n|z_n, \beta)$, la distribución multinomial de probabilidades condicionada sobre el tópico z_n.

Donde cada variable del proceso corresponde a:

- β es la matriz de probabilística de que el documento contenga la palabra w^j dado que discute el tópico z^i, con $B_{ij} = p(w^j = 1|z^i = 1)$.
- θ_d es la distribución de tópicos para el documento d, es decir, el conjunto de probabilidades $\theta_{d,k}$ donde esta corresponde a la probabilidad de que el documento d trate del tópico k.
- z_d son las asociaciones de tópicos para el documento d con $z_{d,n}$ es el tópico asociado a la palabra n-ésima del documento d
- w_d es el conjunto de palabras presentes en el documento d.
- $w_{d,n}$ es la palabra n-ésima del documento d.

A partir de esto, es posible definir el proceso generativo de documentos a través de la distribución conjunta de variables observables y no observables como se define en la ecuación 1. La solución a esta ecuación puede ser obtenida haciendo uso de algoritmos de inferencia estadística como el algoritmo *Sampleo de Gibbs*, los que además de estimar la estructura de tópicos de una colección, permiten inferir la estructura de tópicos presente en otros *corpus* que estén compuestos de documentos que hablen de temas similares a los utilizados para entrenar el modelo.

$$p\left(\beta_{1:K}, \theta_{1:D}, z_{1:D}, w_{1:D}\right) = \prod_{i=1}^{K} p(\beta_i) \prod_{d=1}^{D} p(\theta_d) \left(\prod_{n=1}^{N} p(z_{d,n}|\theta_d) p(w_{d,n|\beta_{1:K}, z_{d,n}}) \right) \tag{1}$$

2.2. Modelos de extracción de opiniones

Con el nacimiento de las redes sociales y la llegada de la Web 2.0, los usuarios comenzaron ser capaces de generar nuevo contenido en la web y también de dar a conocer sus opiniones sobre variados hechos, productos, servicios y cualquier otro tema que sea susceptible de generar un sentimiento o una opinión en ellos.

Para aprovechar esta nuea información que está siendo generada en la web se han desarrollado una serie de metodologías, algoritmos y técnicas para recuperar información desde documentos opinados. Esta nueva rama de la recuperación de la información es llamada *Web Opinion Mining*, la cual tiene como objetivo principal extraer información a partir de las opiniones que se encuentran en los documentos opinados publicados en la web [9].

Los modelos de opinión son utilizados frecuentemente en donde es necesario hacer uso de las opiniones de los usuarios para evaluar u obtener información sobre productos y servicios. En [15] se menciona que los algoritmos de minado de opiniones son utilizados frecuentemente en detección de spam en review de productos, creación y mejoramiento de sistemas de recomendación de productos y servicios o de avisaje online, evaluación de nuevos productos en la web, evaluar el impacto que tienen las reviews en las utilidades de un negocio o un producto, etc.

Una opinión se define como una creencia subjetiva por parte de un sujeto sobre algún objeto, tema o situación en particular, que nace de una interpretación emocional por parte de éste del objeto bajo análisis o una característica de este [6]. Por consiguiente, una opinión es una creencia subjetiva de un *emisor* sobre el *receptor* o una característica de este, y posee una polaridad que señala el tipo de emoción (positiva o negativa) que da paso a la opinión propiamente tal.

Los modelos de extracción o minado de opiniones trabajan sobre *documentos opinados*, los cuales son definidos en [9] como todo documento que contenga una o más oraciones que expresan una opinión. Por lo tanto, se puede decir que los modelos de extracción de opiniones buscan determinar qué tipo de emoción motiva la emisión de una opinión en un documento [6] o que polaridad es la predominante en este [18].

Para dar a conocer un modelo de opiniones es necesario presentar una serie de definiciones que dan sustento a la gran mayoría de los modelos utilizados en la actualidad. Estas definiciones son las que siguen:

- **Objeto**: Un *objeto o* es una entidad que puede ser un producto, un servicio, un individuo, una organización, un evento, etc. descrito por la dupla (T, A) donde T es la jerarquía que describe cada una de las componentes del objeto y A es el conjunto de atributos de este. A su vez, cada componente posee su propio conjunto de sub-componentes y atributos.

- **Opinión**: Una *opinión* sobre una característica f objeto o es una evaluación emocional que realiza un *emisor* sobre este o una característica de él.

- **Emisor**: El *emisor* de una opinión es aquella persona u organización que la expresa.

- **Polaridad**: La *polaridad* de una opinión indica si la opinión es *positiva*, *negativa* u *objetiva*.

Además, en el modelo de análisis de opiniones basado en características, un objeto o se describe como un conjunto de características $F = f_1, f_2, ..., f_n$ donde también se incluye el objeto en cuestión como una característica particular. En este caso, cada característica f_i puede ser descrita por el conjunto de palabras o frases $W_i = w_{i1}, w_{i2}, ..., w_{im}$, donde cada w_{ij} es un sinónimo de la característica f_i; además, f_i también puede ser expresada a través del conjunto de indicadores de característica $I_i = i_{i1}, i_{i2}, ..., i_{iq}$.

Bajo este modelo, un documento d que contiene opiniones es descrito como aquel que contiene opiniones sobre un conjunto de objetos $o_1, o_2, ..., o_q$ emitidas por un conjunto de emisores $h_1, h_2, ..., h_p$. En este caso, cada opinión o_j se enfocan en un subconjunto F_j de características del objeto en cuestión y puede ser clasificada en uno de los siguientes tipos:

- **Opinión directa**: Es la quintupla $(o_j, f_{jk}, oo_{ijkl}, h_i, t_l)$, donde o_j es el objeto sobre el cuál consiste la opinión, f_{jk} es la característica del objeto o_j que está siendo analizada, oo_{ijkl} es la polaridad de la opinión sobre la característica f_{jk}, h_i es el emisor de la opinión y finalmente, t_l es el momento en el cuál h_i expresó la opinión.

- **Opinión comparativa**: Expresa la relación, sea esta de similitud o de diferencia entre dos o más objetos y las preferencias del emisor de la opinión sobre un conjunto común de características entre los objetos.

Toda opinión se basa en las emociones que guían al emisor a emitirla en el momento que este acto sucede. De acuerdo a lo expresado en [9], las emociones son ***sentimientos y pensamientos subjetivos***, y estas se dividen en 6 tipos primarios: *amor, alegría, sorpresa, rabia, tristeza* y *temor*.

Si bien todas las opiniones nacen de una emoción, la manera en que estas son expresadas por el emisor de ellas permite clasificarlas en dos tipos: las opiniones *explícitas*, aquellas en que el emisor expresa claramente la opinión a través de una frase subjetiva; y las *implícitas*, donde la opinión en cuestión es expresada a través del uso de una frase objetiva. Un ejemplo de opinión explícita es *"me encanta el sabor de este helado"* y de opinión implícita es *"la linterna explotó a la semana de haberla comprado"*.

2.2.1. Aplicaciones de los algoritmos de minado de opiniones

Entre las aplicaciones que tienen los algoritmos de minado de opiniones podemos encontrar:

1. **Análisis de reviews de productos**: En [2] se discuten distintas aplicaciones de estos algoritmos en el análisis de reviews de productos, entre ellas se destacan el resumen de opiniones, detectar reviews falsos o spam y la evaluación monetaria de las características de un producto.
2. **Sistemas de recomendación**: En [7] se estudia mejorar sistemas de recomendación de productos a través del uso de las opiniones emitidas por usuarios de estos mismos sistemas.
3. **Política**: En [13] se muestran distintos enfoques para analizar campañas políticas y la percepción de la gente sobre leyes y candidatos políticos.

2.2.2. Algoritmos para extracción de polaridad de opiniones

En la plataforma de detección de tendencias se hace uso de algoritmos de detección de polaridad para determinar qué es lo que se opina en la web sobre los tópicos que son extraídos desde los documentos recuperados. A continuación se dará a conocer las dos afluentes más utilizadas de algoritmos de detección de polaridad en opiniones.

Algoritmos de clasificación a través de aprendizaje supervisado: La mayoría de los algoritmos de aprendizaje supervisado existentes (Naive-Bayes, Support Vector machines, etc.) pueden ser aplicados a la clasificación de polaridad de documentos tal como se muestra en [16, 11].

El algoritmo más utilizado debido a su simplicidad es un clasificador Naive-Bayes, el cual busca obtener las probabilidades de que un documento d posea la polaridad p $\Pr(p \mid d)$. Este tipo de clasificador obtiene estas probabilidades al resolver el siguiente problema de maximización: $arg\,\max_{p \in P}\{\Pr(p \mid d)\}$.

Los clasificadores de Naive-Bayes hacen uso de la regla de Bayes para poder simplificar el problema de maximización que deben resolver:

$$p_d = arg\,\max_{p \in P}\left\{\frac{\Pr(d \mid p) \cdot \Pr(p)}{\Pr(d)}\right\} \tag{2}$$

Debido a que sólo se busca conocer la probabilidad de que un documento tenga una polaridad y no obtener un puntaje específico para el nivel de polaridad que posee, el denominador de la ecuación 2 puede ser eliminado. Esto junto con el hecho de que uno de los supuestos del clasificador de Naive-Bayes es la independencia condicional entre todas las polaridades, se puede decir que:

$$\Pr(d \mid p) = \prod_{i=1}^{m} \Pr(w_i \mid p) = \prod_{i=1}^{m} \frac{\#(w_i, p)}{\#(w_i)} \tag{3}$$

Con $\#(w_i, p)$ el número de veces que la palabra w_i se ha encontrado en documentos de polaridad p en el conjunto de entrenamiento y $\#(w_i)$ el número de veces que la palabra w_i aparece en este último. Para evitar que existan probabilidades 0, se realiza un proceso llamado "suavización de Laplace"que consiste en lo siguiente:

$$\Pr(d \mid p) = \prod_{i=1}^{m} \frac{\#(w_i, p) + 1}{\#(w_i) + m} \tag{4}$$

Con estas ecuaciones basta resolver el problema de maximización planteado para obtener las probabilidades de que cada documento posea una polaridad en particular.

En general, las características utilizadas por los algoritmos de aprendizaje supervisado se dividen en las siguientes categorías:

- *Frecuencia y presencia de términos*: Si bien el uso de frecuencia de aparición de términos, por ejemplo a través del modelo *tf-idf*, en la recuperación de la información siempre ha sido de mucha utilidad, en [16] se muestra que en el caso de la extracción de opiniones desde documentos la *presencia* de un término es más importante que la frecuencia con que este aparece.

- *Partes del discurso*: Los adjetivos han sido utilizados con frecuencia [11] en el uso de algoritmos de aprendizaje supervisado ya que existe una alta correlación entre la presencia de adjetivos en una oración y la subjetividad de esta.

- *Sintáxis*: En [12] se hace uso de la relación entre las palabras como características en algoritmos de aprendizaje supervisado.

Algoritmos de clasificación a través de aprendizaje no supervisado: En [19] se propone un algoritmo de aprendizaje no supervisado para la clasificación de polaridad de documentos que se compone de tres etapas:

1. Se extraen todas las frases con verbos o adjetivos, ya que tal como se menciona en [11], estas partes del discurso se han mostrado muy útiles a la hora de detectar opiniones en documentos. Sin embargo, a pesar de que un adjetivo por si solo puede demostrar subjetividad, puede que no exista la información suficiente para determinar la polaridad de la opinión. Debido a esto, este algoritmo trabaja con pares de palabras, una de ellas siendo un adjetivo y la otra una palabra contextual que facilita la determinación de la polaridad de la oración en cuestión. Estos pares de palabras son extraídos siempre y cuando, considerando las dos palabras y la que les sigue, correspondan a alguno de los patrones conocidos.

2. Se estima la polaridad de las frases extraídas, haciendo uso de la métrica de dependencia estadística entre términos llamada *pointwise mutual information* (PMI) que se presenta en la ecuación 5

$$PMI(w_1, w_2) = log_2 \left(\frac{\Pr(w_1 \wedge w_2)}{\Pr(w_1) \Pr(w_2)} \right) \tag{5}$$

   Luego, la polaridad de una frase puede ser calculada basándose en el nivel de asociación entre ella y las palabras de referencia *pobre* y *excelente* a través de la ecuación 6

$$oo(frase) = PMI(frase, "excelente") - PMI(frase, "pobre") \tag{6}$$

3. Finalmente, el algoritmo calcula la polaridad *oo* promedio de todas las frases en el documento y lo clasifica dependiendo de si el promedio es positivo o negativo.

Algoritmos basados en lexicones de opinión: Los algoritmos basados en lexicones de opinión son los algoritmos más sencillos y a su vez los que buscan ser de uso más general debido a que la información utilizada para determinar la polaridad de una opinión no está restringida a ningún dominio en particular. Un lexicón es un conjunto de palabras rotuladas con polaridad de sentimientos, es decir, cada palabra perteneciente al lexicón tiene asociado un puntaje de polaridad.

Estos algoritmos trabajan bajo la hipótesis de que una palabra es considerada la unidad elemental de una opinión y por lo tanto la polaridad de una opinión puede reconstruirse a partir de la polaridad de cada una de las palabras que la componen. Ejemplos de algoritmos que hacen uso de lexicones para determinar la polaridad de una opinión se pueden encontrar en [14, 16].

En relación al minado de opiniones desde documentos de microblogging, Kouloumpis et al. dan a conocer en [8] que los algoritmos de basados en lexicones pueden dar buenos resultados.

El lexicón utilizado por la plataforma de detección de tendencias es *SentiWordNet* el cual está disponible públicamente para ser usado en este tipo de aplicaciones de minado de opiniones.

Cada palabra presente en un lexicón tiene asociado un puntaje por cada polaridad positiva, negativa y objetiva que representan el aporte de esta palabra para la polaridad de una opinión. En el caso de *SentiWordNet* [14] se tiene que cada palabra tiene asociado sólo los puntajes de polaridad positiva w^p y negativa w^n, y además el puntaje de objetividad $w^o = 1 - w^p + w^n$.

Los algoritmos basados en lexicones de opinión hacen uso de las siguientes metodologías para reconstruir la polaridad de la opinión contenida en un documento a partir de sus palabras:

- **Conteo de palabras**: los puntajes de polaridad de un documento se obtiene a través de la fracción de palabras cuya que posee una polaridad predominante p. En este caso, una palabra será considerada de una polaridad p si su mayor puntaje es el de aquella polaridad.
- **Promedio de palabras**: En un algoritmo de promedio de palabras, el puntaje asociado a una polaridad p es el promedio de los valores de polaridad p de todas las palabras presentes en el documento.

A partir de estas metodologías básicas se pueden realizar diversas variaciones tales como: modificar los puntajes de cada palabra en base al conjunto de palabras que la rodean en el documento, hacer uso de las negaciones y la capitalización, y finalmente incorporar al puntaje la existencia de intensificadores y disminuidores de adjetivos.

2.3. Modelos de detección de Tendencias

Los modelos de detección de tendencias buscan modelar el comportamiento de los tópicos tanto desde el punto de la cobertura que este tenga en la Web, como también la percepción que los usuarios de esta tengan sobre él. Por esto, los modelos de detección de tendencias van un paso más allá que los modelos de *topic tracking*, buscando analizar también la percepción que tiene la sociedad del tópico en particular y en cómo ambas componentes se relacionan para convertir un tópico en una tendencia.

En los últimos años se han realizado variados acercamientos a la detección de tendencias en la Web. Aplicaciones enfocadas en el uso de *key-words* para extraer tendencias en Web-usage mining se presentan en [21], política [13], finanzas [17] y sistemas de recomendación [7]. Sin embargo, la contribución de

este trabajo se asemeja más a metodologías genéricas que van más allá de un dominio en particular, como la presentada en [20], cuyo foco es principalmente el cómo construir una plataforma de detección de tendencias sobre una arquitectura de *cloud computing*, y no en como recuperar la información necesaria ni como decidir si es que un tópico discutido en un conjunto de documentos a lo largo del tiempo refleja una tendencia.

3. Detección de Tendencias en la Web

En la actualidad, el análisis de tendencias se ha abordado tradicionalmente a partir de encuestas, las cuales poseen un alto contenido de subjetividad, y puede conducir a errores significativos a la hora de representar los hechos que sucederán en el futuro. Estos errores pueden darse debido al contexto en que las encuestas son realizadas, las motivaciones que la gente tiene a la hora de responder y otros factores exógenos al instrumento en si.

Por otro lado, en las redes sociales, las opiniones consignadas por sus usuarios son una expresión neta de sus sentimientos. Al ser estas no obligadas ni apresuradas, es posible complementar los resultados obtenidos a través de las encuestas con un análisis de estas, reduciendo el ruido producido debido a los factores previamente mencionados.

Para comprobar la hipótesis mencionada en la primera sección de este artículo, se diseñó una plataforma de detección de tendencias que analiza la información presente en la Web en dos ejes: el primero se enfoca en el análisis de eventos, que viene dado por los documentos presentes en los sitios de noticias; y aquel que trata de los sentimientos que expresan los usuarios de las redes sociales sobre aquellos eventos. Cabe destacar que esta plataforma hace uso de técnicas existentes de algoritmos de recuperación de la información y también modelos de tópicos y minado de opiniones para modelar cómo los tópicos se comportan a lo largo del tiempo en busca de identificar tendencias en la Web.

Inicialmente se describirá el enfoque utilizado para recuperar noticias y extraer qué tópicos están siendo discutidos en la blogosfera y en los sitios de noticias. Posteriormente se dará a conocer la metodología para extraer las opiniones a partir de los documentos publicados por los usuarios de las redes sociales, y finalmente la metodología utilizada para juntar ambos conjuntos de información con el fin de identificar tendencias en la Web.

3.1. Plataforma de Detección de Tendencias

Tal como se menciona en la sección de trabajo relacionado, los modelos de tendencias no sólo buscan detectar qué tópicos se discuten a lo largo del tiempo en un corpus de documentos, también tienen como objetivo analizar

las reacciones sociales que provocan estos tópicos . En el caso de este trabajo de investigación, se acotan a las opiniones consignadas por los usuarios de redes sociales a lo largo del periodo de análisis. Así, la plataforma propuesta debe estar formada por dos pilares fundamentales: la detección de tópicos a lo largo del tiempo, y el análisis de dichos tópicos en las redes sociales a través del minado de las opiniones que les competen.

3.1.1. Minado de noticias

Se considera que una fuente de documentos presente en la web es un *feed* si cada elemento que esta contenga es desplegado de manera cronológica y pertenecen todos una misma temática. Si una fuente de documentos dispone de un punto de acceso donde se puedan recuperar cada uno de los documentos existentes en ella se dice que es un *feed sindicable*, un ejemplo de esto son todos aquellos sitios web que tienen la opción de suscribirse a su contenido a través de RSS.

Una limitante a considerar a la hora de trabajar con feeds sindicables es que el conjunto de documentos presentes cuando se accede a esta depende del tiempo. Esto implica que el conjunto de documentos $\{d_i^F\}_{i\in\mathbb{N}}$ que se obtienen al solicitar todos los documentos desde la fuente F se ve limitada por el momento t en el cual se realice esta petición. En este caso, se define $\{d_i^{F^t}\}_{i\in\mathbb{N}}$ como el conjunto de documentos recuperados desde una fuente F en un instante de tiempo t. Además, se define $\{F_i\}_{i\in\mathbb{N}}$ como el conjunto de *feeds* que recorrerá el módulo de recuperación de documentos a través de su *crawler* para alimentar el módulo de extracción de tópicos.

Para este proyecto, sólo se trabajará con feeds sindicables, por lo que, en base a lo anterior es posible definir un algoritmo de recuperación de documentos a partir de una lista de fuentes $\{F_i\}_{i\in\mathbb{N}}$ sindicables (sean estas *RSS* o *Atom*) como se describe en el algoritmo 3.1:

Algoritmo 3.1: Recuperación de documentos

Data: $\{F_i\}_{i\in\mathbb{N}}$, t
Result: $\bigcup_i \{d_j^{F_i^t}\}_{j\in\mathbb{N}}$

```
documents := [];
for i ← 1 to ‖{F_i}_{i∈ℕ}‖ do
    document ← retrieveDocument(F_i);
    documents ← documents ⋃ document;
return documents;
```

Para extraer qué tópicos se discuten a lo largo del tiempo en la Web, se propone utilizar un enfoque basado en modelo de tópicos, debido a que permiten de manera directa obtener las *keywords* necesarias para posteriormente extraer

las opiniones presentes en las redes sociales, y además, permiten monitorear la evolución de los tópicos a lo largo del tiempo. El hacer uso de técnicas de *topic tracking* o *topid detection* no es recomendable debido a las limitaciones que estos imponen para la posterior recolección de opiniones asociadas a cada tópico.

Una vez recuperados los documentos desde los sitios de noticias, se procede a utilizar el modelo LDA para recuperar qué tópicos se están tratando en ellos. Este modelo permite, dada una colección de documentos $\{d_i\}_{i=1...\mathbb{N}}$, obtener un conjunto de tópicos t asociados a documentos, los cuales están descrito por la probabilidad $P(topic = t|document = d)$ de que un documento d pertenezca al tópico t y además, para cada tupla (w, t) la probabilidad $P(topic = t|word = w)$ de que una palabra w describa al tópico t. Así, es posible obtener los tópicos que se tratan a lo largo del tiempo en los feeds que se están minando y las palabras que los describen para luego utilizar esta información con el fin de recuperar documentos opinados desde las redes sociales.

Para cada periodo t_i, se toman todos los documentos de los dos periodos anteriores t_{i-1}, t_{i-2} y se entrena un nuevo modelo LDA con estos. Luego, para los documentos del periodo t se realiza inferencia con el modelo LDA sobre estos para descubrir el modelo de tópicos subyacente en estos.

Una vez que se tengan los documentos de los periodos t_i, t_{i-1}, t_{i-2}, es posible enlazar dos tópicos T y T', con vectores de probabilidades de palabras $\vec{w}_T$ y $\vec{w}_{T'}$ través de una función de distancia de tópicos que se define como sigue:

$$d(T, T') = \sum_{w \in \vec{w}_T} \sum_{\vec{w}_{T'}} w_i - w_j \tag{7}$$

Y luego, dado toda dupla T y T' de tópicos, se enlazan sí y sólo si el resultado la función $d(T, T')$ está bajo un umbral ϕ que se define a la hora de comenzar el análisis.

3.1.2. Minado de Opiniones

Una vez que se han extraído los tópicos a partir de los documentos presentes en sitios de noticias, se procede a extraer las opiniones sobre cada uno de ellos en las redes sociales a través del uso de modelos de minado de opiniones. Con esto es posible es posible obtener un puntaje de opinión para cada tópico a lo largo del tiempo a partir de una colección de documentos. La metodología a utilizar es la siguiente:

Para definir qué documentos serán recuperados desde las redes sociales, se tiene el algoritmo 3.2, que dado un tópico T obtiene todos los n-gramas de largo n que caracterizan a ese tópico en particular en el periodo t. El parámetro N

consiste en la cantidad de palabras que deben ser utilizadas para la obtención de los unigramas que describen al tópico, además, el método $T.\texttt{words}(t, N)$ obtiene las N palabras más relevantes del tópico T en el periodo t.

Algoritmo 3.2: Método `generateQueries`

Data: T, t, n, N
Result: $\{query_i\}_{i \in \mathbb{N}}$

```
queries := [];
words = T.words(t, N);
forall p ∈ permutaciones(words, n) do
    queries.append(p);
return queries;
```

Luego, para cada tópico T, se obtienen todas las queries que le correspondan, y se obtienen documentos opiniones en las redes sociales que se determinen utilizando sus APIs. En el caso particular de este experimento, sólo se trabajará con la red social de microblogging Twitter. Para cada documento, se procede a obtener su polaridad de la siguiente manera:

Algoritmo 3.3: Clasificación de documentos opinados

Data: $\{d_i\}_{i=1...\mathbb{N}}$
Result: $\{\vec{d_i}\}_{i \in \mathbb{N}}$

```
documents := [];
for i ← 1 to ‖{d_i}_{i=1...N}‖ do
    d_i ← polaridad(d_i);
    documents.append(d_i);
return documents;
```

3.1.3. Visualización de Tendencias

Una vez obtenidas las noticias y las opiniones relacionadas a los tópicos en discusión, se procede a generar un gráfico como el presentado en la figura 1 que representa el comportamiento de cada tópico a lo largo del tiempo. En donde las barras corresponden a la cantidad de documentos en los cuales se hace mención del tópico para cada periodo de tiempo, y la opinión de los usuarios de las redes sociales a lo largo del tiempo con respecto a este se representa como una línea.

3.2. Diseño del experimento

Sobre una temática en particular, se visitará periódicamente un conjunto de 20 sitios de noticias que publiquen documentos sobre esta, y se ejecutará la metodología presentada a lo largo de un mes.

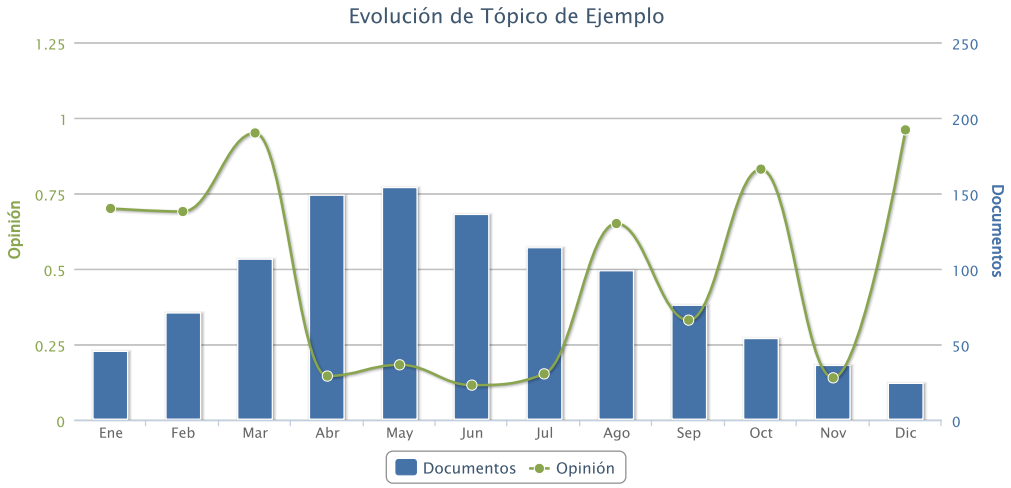

Figura 1: Ejemplo de gráfico por tópico

3.2.1. El entorno

Los sitios de noticias. En cuanto a los sitios, se requiere satisfacer tres requerimientos: en primer lugar, debe tener una frecuencia de publicación adecuada. Además, la cantidad de documentos por sitio no puede ser excesivo y, por último, estos deben estar en inglés para facilitar el procesamiento de los documentos.

El tema a analizar. El tema a analizar debe ser capaz de generar discusiones en las redes sociales, o al menos muestras de apreciación o desagrado, ya que de manera contraria no será posible realizar la última etapa del proceso de detección de tendencias y por lo tanto, el experimento se verá invalidado.

3.2.2. Captura y transformación de datos

Sitios de noticias. Una vez elegidos los sitios, estos serán visitados periódicamente para recuperar los artículos publicados en ellos. Cada artículo será almacenado con su contenido original, y para su procesamiento se procederá a remover todo el contenido que no sea texto plano (por ejemplo, *tags* de **html**) y todas las stopwords que se encuentren.

Opiniones. Al igual que los artículos recuperados de los sitios de noticias, estos serán almacenados tal como fueron extraídos desde su fuente.

3.3. Soluciones existentes para detección de tendencias

En el ámbito académico, múltiples investigaciones [10, 1, 5] han abordado la detección de tendencias en la web, principalmente en las redes sociales, destacándose entre ellas dos tipos distintos, aquellas que tienen como objetivo detectar de manera temprana aquellos tópicos que serán tendencia en el corto

plazo, y las que buscan detectar aquellos tópicos que están siendo tendencia y su presencia va en aumento a lo largo del tiempo.

En aplicaciones comerciales, la plataforma web *NewsWhip*[1] ofrece prestaciones similares a las presentes en la plataforma de detección de tendencias presentada, sin embargo, su enfoque es lograr ser un agregador de noticias con características sociales, como la medición de menciones en las redes sociales de una noticia en particular o el análisis de noticias de una empresa en particular en la web. Además, *NewsWhip* ofrece la herramienta *Spike*, que permite a los generadores de contenido analizar cómo sus noticias se esparcen por la web.

La empresa *Sysomos*[2] se enfoca en monitorear las redes sociales en búsqueda de información relevante para una empresa en particular, sin embargo, no hacen uso de la información presente en las noticias y no tienen como objetivo hacer un análisis extenso de las tendencias en la Web, si no monitorear las conversaciones que se están realizando en las redes sociales.

Otra iniciativa que busca detectar tendencias en la Web es Google Trends, la cual toma un enfoque distinto a los ya mencionados al analizar el comportamiento de búsqueda de los usuarios de su motor de búsqueda, sin embargo, no hacen uso de los datos presentes en su red social Google+ para complementar las tendencias obtenidas con información sobre las opiniones de la gente sobre ellas.

[1] `http://www.newswhip.com/`
[2] `http://www.sysomos.com/`

4. Aplicación del experimento y análisis de resultados

4.1. Captura de datos

Para recuperar los documentos existentes en los sitios de noticias o blogs que se analizaron, se implementó un *crawler* hecho en Java capaz de parsear y recuperar información desde fuentes *RSS*. Para cada fuente *RSS*, se solicita periódicamente la lista de artículos presente en ella, y en caso de que se encontraran nuevos elementos en relación a la última extracción de documentos se procede a almacenar esta diferencia en la base de datos. En el caso de los documentos opinados recuperados desde las redes sociales, también se desarrolló un *crawler* en Java para recuperar los documentos opinados asociados a un tópico en particular.

4.2. Aplicacion del Modelo de Detección de Tendencias

4.2.1. Entorno

La temática escogida: Los experimentos se desarrollaron con el fin de analizar lo sucedido en la temática de la tecnología y sus ramificaciones, en particular, se enfocó el estudio sobre noticias y opiniones en inglés. Ambas elecciones se realizaron en base a la alta cantidad de información disponible sin importar el periodo en el cual se realizara en el estudio.

Los sitios analizados: Se escogió de manera manual una muestra de 20 blogs o sitios de noticias en inglés que traten la temática de la tecnología. Cada uno de estos debe disponer de su contenido en formato RSS para una más fácil recuperación de sus artículos.

El periodo de análisis: Para el desarrollo del análisis se analizaron sitios de noticias entre Abril del 2011 y Enero del 2012, analizando los tópicos tratados por ellos en dicho periodo.

4.2.2. Experimentos

Como primer experimento, se aplicó la metodología presentada en el entorno previamente descrito, a partir del cual se procedió a analizar los tópicos extraídos y los gráficos temporales para cada uno de estos, y se determinó si la información presentada en ellos correspondía a lo que se podía observar a partir del análisis de los hechos ocurridos en este periodo. Por otro lado, el

experimento	10	20	30
primero	66 %	58 %	49 %

Tabla 1: *Precision*

experimento	10	20	30
primero	37 %	46 %	59 %

Tabla 2: *Recall*

segundo experimento consistió en el análisis experto de estos gráficos para ver si dichos tópicos podían ser categorizados como tendencias.

4.3. Resultados Obtenidos

Luego de analizar los sitios de noticias previamente elegidos durante el periodo de análisis con un número variable de tópicos por periodo, y considerando una semana por iteración de la metodología, se encontró que la cantidad de tópicos extraídos por el modelo LDA en cada periodo que ofrecía mejores resultados correspondía a 10 y además, se determinó hacer uso de periodos de 7 días de largo.

4.3.1. *Precision y recall*

En la tabla 1 se muestra la precisión lograda en el primer experimento para las tres cantidades de tópicos por semana que fueron seleccionadas. Se puede observar que a medida que la cantidad de tópicos por periodo aumenta, la *Precision* del algoritmo disminuye, ya que a medida que esta variable aumenta, la granularidad del modelo LDA aumenta, provocando que un tópico descubierto por inspección sea dividido en dos tópicos más pequeños pero altamente relacionados. Este suceso ocurre en todo dominio que se quiera analizar, sin embargo, a medida que el dominio bajo análisis es más amplio, la cantidad óptima de tópicos por periodo aumenta. Por lo tanto, es necesario ajustar el modelo dependiendo del dominio bajo análisis.

En la tabla 2, se observa que el *Recall* aumenta a medida que la cantidad de tópicos por periodo aumenta. Esto se debe a que si bien hay una mayor fragmentación de macrotópicos, se incluyen tópicos pequeños independientes que son absorbidos por ellos cuando la cantidad de tópicos por periodo disminuye.

En el caso del segundo experimento, se observó que un 63 % de los tópicos observados tuvieron un comportamiento similar al esperado por los expertos consultados, lo que indica, que a pesar de que la herramienta es propuesta como un apoyo a la detección de tendencias, esta realiza un buen trabajo en modelar el comportamiento de los tópicos a lo largo del tiempo.

5. Conclusiones

En este trabajo de investigación se demostró que es posible hacer uso de una herramienta de detección de tendencias basada en datos presentes en la web, para mejorar la calidad de la información provista por medios tradicionales de detección de tendencias como lo son las encuestas de opinión.

Para lograr este resultado se realizó un amplio estudio de cuáles de los datos originados en la web pueden complementar la información presente en los medios tradicionales, junto con los modelos matemáticos que se usan para describir tópicos en colecciones de documentos y la manera en que los usuarios de la web expresan sus opiniones en las redes sociales.

Si bien esta metodología es un complemento para los medios tradicionales, una de sus limitaciones es que la demografía de los usuarios de Internet, y aquellas personas accesibles a través de encuestas no siempre coinciden, por lo que si se desean realizar estudios enfocados en ciertos sectores de la población es posible que esta metodología no logre aportar suficiente valor. Por otro lado, al realizar el análisis de los datos de manera periódica, no es posible dar alerta temprana de sucesos que ocurren en el día a día. Por ello, los resultados entregados por esta herramienta deben ser considerados como un apoyo a decisiones de negocio enfocadas en un mercado en particular y también como un complemento a metodologías tradicionales de detección de tendencias.

Como trabajo futuro, se plantea considerar nuevas técnicas de minado de opiniones que se especialicen en documentos obtenidos desde sitios de microblogging y además características de estos como la ironía y los acrónimos de expresiones populares. Por otro lado, se plantea modificar el modelo de tópicos usado para que sea capaz de detectar reapariciones de tópicos después de un tiempo prolongado. Finalmente, se propone la evaluación del impacto de implementar un sistema de alerta temprana.

Agradecimientos: Este trabajo fue parcialmente financiado por el Proyecto FONDEF project D10I- 1198: WHALE: Web Hypermedia Analysis Latent Environment y por el Instituto Milenio Sistemas Complejos de Ingeniería (ICM: P-05-004-F, CONICYT: FBO16).

Referencias

[1] F. Alvanaki, M. Sebastian, K. Ramamritham, and G. Weikum. Enblogue: emergent topic detection in web 2.0 streams. In *Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data*, SIGMOD '11, pages 1271–1274, Athens, Greece, 2011. ACM.

[2] Nikolay Archak, Anindya Ghose, and Panagiotis G. Ipeirotis. Show me the money!: deriving the pricing power of product features by mining consumer reviews. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '07, pages 56–65, San Jose, California, USA, 2007. ACM.

[3] David M. Blei and John D. Lafferty. Correlated topic models. In *NIPS*, 2005.

[4] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, March 2003.

[5] Irena Pletikosa Cvijikj and Florian Michahelles. Monitoring trends on facebook. In *Proceedings of the 2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing*, DASC '11, pages 895–902, Sydney, Australia, 2011. IEEE Computer Society.

[6] T Damer. *Attacking faulty reasoning: a practical guide to fallacy-free arguments*. Wadsworth/Cengage Laerning, Australia Belmont, CA, 2009.

[7] Shay David and Trevor John Pinch. Six degrees of reputation: The use and abuse of online review and recommendation systems. *First Monday*, July 2006. Special Issue on Commercial Applications of the Internet.

[8] Efthymios Kouloumpis, Theresa Wilson, and Johanna Moore. Twitter sentiment analysis : The good the bad and the omg ! *Artificial Intelligence*, 70(2):538–541, 2011.

[9] Bing Liu. Sentiment analysis and subjectivity. In Nitin Indurkhya and Fred J. Damerau, editors, *Handbook of Natural Language Processing, Second Edition*. CRC Press, Taylor and Francis Group, Boca Raton, Florida, USA, 2010. ISBN 978-1420085921.

[10] Michael Mathioudakis and Nick Koudas. Twittermonitor: trend detection over the twitter stream. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data*, SIGMOD '10, pages 1155–1158, Indianapolis, Indiana, USA, 2010. ACM.

[11] T. Mullen and N. Collier. Sentiment analysis using support vector machines with diverse information sources. In *Proceedings of Conference on Empirical Methods in Natural Language Processing*, volume 4 of *EMNLP '04*, pages 412–418. ACL, 2004.

[12] V. Ng, S. Dasgupta, and SM Arifin. Examining the role of linguistic knowledge sources in the automatic identification and classification of

reviews. In *Proceedings of the COLING/ACL on Main conference poster sessions*, pages 611–618. Association for Computational Linguistics, 2006.

[13] B. O'Connor, R. Balasubramanyan, B.R. Routledge, and N.A. Smith. From tweets to polls: Linking text sentiment to public opinion time series. In *Proceedings of the International AAAI Conference on Weblogs and Social Media*, ICWSM '10, pages 122–129. AAAI Press, 2010.

[14] Bruno Ohana and Brendan Tierney. Sentiment classification of reviews using sentiwordnet. *Discovery*, page 13, 2009.

[15] Bo Pang and Lillian Lee. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1–135, January 2008.

[16] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing - Volume 10*, EMNLP '02, pages 79–86, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.

[17] V. Sehgal and C. Song. Sops: stock prediction using web sentiment. In *Data Mining Workshops, 2007. ICDM Workshops 2007. Seventh IEEE International Conference on*, ICDMW '07, pages 21–26, Omaha, Nebraska, USA, 2007. IEEE Computer Society.

[18] Edison M. Taylor, Cristián Rodriguez, Juan D. Velásquez, Goldina Ghosh, and Soumya Banerjee. Web opinion mining and sentiment analysis. In Juan D. Velásquez, Vasile Palade, and Lakhmi C. Jain, editors, *Advanced Techniques in Web Intelligence-2*, pages 105–126. Springer, 2012.

[19] Peter D. Turney. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 417–424, Philadelphia, Pennsylvania, 2002. Association for Computational Linguistics.

[20] Athena Vakali, Maria Giatsoglou, and Stefanos Antaris. Social networking trends and dynamics detection via a cloud-based framework design. In *Proceedings of the 21st international conference companion on World Wide Web*, WWW '12 Companion, pages 1213–1220, New York, NY, USA, 2012. ACM.

[21] Juan D. Velásquez. Web site keywords: A methodology for improving gradually the web site text content. *Intelligent Data Analysis*, 16(2):327–348, 2012.

Sentiment Polarity of Trends on the Web Using Opinion Mining and Topic Modeling

Rodrigo Dueñas-Fernández*, Gaston L'Huillier† and Juan D. Velásquez*
*Department of Industrial Engineering
Universidad de Chile, República 701- P.O. Box: 8370439 Santiago, Chile
†Groupon Inc.
3101 Park Blvd., Palo Alto, CA., USA
rduenas@ing.uchile.cl, gaston@groupon.com, jvelasqu@dii.uchile.cl

***Abstract*—Since the beginning of human society as we know it today, there has been an intrinsic need to discover the unknown, to predict events that could happen, and the mechanics on how it behaves and evolve over time. To approximate a solution to this need, this paper introduces a novel approach for modeling trends based on the information available on the Web through the usage of Opinion Mining and Topic modeling. This methodology was ran over a set of 20 feeds during a period of 8 months. The main result is that, given the obtained F-Measure of 0.56 for the detection of significant events, this methodology is a feasible model of how trends could be represented by the information available on the Web.**

***Keywords*-Trend Detection, Web Opinion Mining, Topic modeling,**

I. Introduction

Nowadays, the most common method for inferring what are the trends of any given topic is to run surveys over a sparse set of individuals. Despite the fact that is commonly used, this method comes with some limitations. For example, given the nature of the interviewer or the channel used to perform the interview, the results of the interview could change dramatically [1]. Therefore, there is a need to complement traditional methods with new techniques that allow to gather information from the Web, where there is a vast amount of people freely expressing their opinion [2].

Detecting trends and their polarity presents many difficulties that need to be overcomed in order to propose a solution. Even if there was an easy way to represent an opinions posted over time by Web users about a given aspect of an entity or topic, there would not be a perfect solution to uniquely represent such feedback about an aspect of an entity on the Web. Furthermore, given there is no unique representation of an aspect of an entity, linking them to a opinion expressed by a Web user is not straightforward.

To the best of our knowledge, there are no unified methodology that solves every problem mentioned above. However, each problem on its own has been tackled by different disciplines. First, information retrieval field has presented several techniques that allow to retrieve and process relevant documents. In terms of detecting the topics of such documents and the aspects for which opinions have been expressed, text mining and natural language processing communities have developed a vast set of models to represent which are the topics across a collection of documents [3]. Finally, Web opinion mining field has presented several approaches to represent the polarity of documents posted by Web users [4]. The main contribution of this work is to integrate all previous disciplines into one unified framework.

This paper is organized as follows: in section II a brief summary of related research is provided. Section III describes the proposed methodology for detecting trends on the Web is described. Section IV outlines the experiments performed with the proposed methodology and Section V provides some conclusions and suggests future research.

II. Related Work

Several approaches for determining the trends in the Web have been previously proposed by different authors. Applications in the analysis of trends in politics are presented in [5], [6], finance [7] and recommendation systems [8]. However, generic frameworks that goes beyond a singular application domain are closer to the contribution of this work. In [9], the focus is mostly in how to build a trend detection framework in a cloud computing architecture, rather than extending the work in terms of how to retrieve the data and deciding whether a topic presented in several documents over time reflects a trend or not.

In terms of information retrieval, there is a vast literature that provides some insights about the different types of data and how this data should be handled [10]. Some specific information retrieval frameworks for the detection of trends in blogging [11], microblogging (e.g. Twitter) [12], social networking sites (e.g. Facebook) [13]. Once the data has been captured and stored, the textual information has to be processed in such way that the underlying patterns are extracted for further usage. In this domain, keyword-based analysis approaches have been proposed in the specific context of Web usage mining [14]. However, one of most relevant techniques used in recent years are topic models [15].

A topic model can be considered as a probabilistic model that relates documents and words through variables which represent the main topics inferred from the text itself. In this context, a document can be considered as a mixture of topics, represented by probability distributions which generate the words in a document given these topics. The inferring process of the latent variables, or topics, is the key component of this model, whose main objective is to learn the distribution of the underlying topics from text in a given corpus of text documents. A main topic model is the latent Dirichlet allocation (LDA) [16]. LDA is a Bayesian model in which latent topics of documents are inferred from estimated probability distributions over a training data set.

Opinion mining and sentimental analysis is known as a field whose objective is that given a collection of opinionated documents, determine the opinion orientation (positive, negative, and objective) from a particular aspect of an entity of an opinion holder at a given time [17]. In terms of trend detections, this task is fundamental so to identify from the documents whether the trending topics are being generated with a certain opinion orientation. In this work, the opinion mining steps will be focused in the usage of lexicon-based algorithms. Algorithms that are based on the use of lexicons can be found in [1], [2], which according to the research presented in [18] can return good information in the context of mining opinions of documents retrieved from microblogging sites.

III. A Methodology for Trend Detection on the Web

In this chapter, the methodology proposed to detect trends on the Web is presented. First, the definition of the problem to solve and every term used through this paper are detailed. Next, some of the main text analysis techniques used during the development of the proposed methodology for detection of trends on the Web are discussed. Finally, the methodology itself and the main contribution of this work is described.

A. Problem Definition and General Notation

In the following, the term *Trend* will be presented together with its ontological and linguistic representation. In this context, a trend will be defined as a given event whose impact on a system as a whole, is above the average over a certain period of time.

The problem of detecting trends on the Web is described. Several research areas are focused in modeling the so called collective behavior in order to, for example, predict how important events will develop, which politic will win a debate, which football team will win a match and so on.

Even though existing methodologies to predict trends and monitor their evolution over time have been successfully applied to a vast variety of problems, there's a huge amount of information not being used, created by users on the Internet, where the act of expressing one's opinion or feelings is not restrained by the common issues that are found in the standard methodologies such as limited time and biased answers based on the person running the interview.

The objective of this research is to tackle what will be called as the *Trend Detection Problem*, which is defined as:

Definition 1: **Trend Detection Problem**: Given a set of topics, determine if the way they behave over time makes them qualify as a trend.

In this work, a *factual document* is a document that contains no opinion whatsoever and refers to one or more events. On the contrary, the term *opinionated document* refers to any opinionated document whose subject is an event. Example of these type of documents are tweets, opinion columns in journals, among other personal opinions by a given Web user.

As the detection of trends in most useful scenarios is always framed within a certain domain of knowledge, a similar approach will be taken in our methodology, where the set of websites to be crawled for documents is defined beforehand and they're expected to be bounded within an specific domain of knowledge.

The proposed methodology for detecting trends on the Web consists of three main steps that are executed periodically and then complemented by the visualization of the extracted data. These three steps are:

1) Crawl every site in the defined set of feeds for factual documents.
2) Infer the underlying topic structure for the documents retrieved during the whole period and link them with the ones extracted in past periods.
3) Retrieve opinionated documents and extract sentiment information for every topic being discussed in the current period.

B. Detecting topics towards an opinion mining analysis

Based on the given the definition of the TD problem, to detect trends on the Web the first step that needs to be accomplished is to extract the topics that are being discussed on the subset of the web containing factual documents. In order to do so, and being able to gather the needed data from social network websites to perform a sentiment analysis, a crawling algorithm is used the documents retrieved from such websites.

Given a set of feeds, a simple crawling algorithm can be used to retrieve the raw documents from each feed. Documents retrieved by this crawling algorithm are stored as raw data together with the extracted metadata that could be extracted from the feed that it came from. Some of the information present as metadata in these feeds are categories and labels used on the site to classify content, author, language, original publication date, among other type of information relevant to the particular analysis.

Once the documents are retrieved from each feed, an LDA[16] model is used to extract the underlying structures

for the topics that are being discussed on them. This model allows, given a collection of documents $\{d_i\}_{i=1...\mathbf{N}}$, obtain a set of topics $\{t_i\}_{i=1\in\mathbf{N}}$ described by the probability $P(topic = t|document = d)$ for a document d to discuss topic t and, for each pair of words and topics (w, t), the probability $P(topic = t|word = w)$ for a word w to describe a topic t.

To achieve a representation of how topics evolve over time, is necessary to extract a set of topics for each period t_i and link these with the topics of the prior period and so on. One of the limitations of the LDA model is that it doesn't correlate topics over time, therefore, it's mandatory to create a way to correlate topics extracted during a period t with the topics extracted from documents retrieved on past periods. The approach proposed for this research is the following:

1) For every period t, collect the documents from the two preceding periods t_{i-1}, t_{i-2} and use them as train data for a new LDA.
2) Then, using the trained model a Bayesian inference is performed over the set of documents retrieved in period t, to discover the underlying topic structure in the collection.

Once every document published on periods t_i, t_{i-1}, t_{i-2} is retrieved and the topic structure that represents the documents retrieved in t_i is inferred, is possible to link two topics T y T', with corresponding word-topic probability vectors $\vec{w}_T$ and $\vec{w}_{T'}$ making use of a distance function defined as shown in equation 1:

$$d(T, T') = \sum_{w_i \in \vec{w}_T} \sum_{w_j \in \vec{w}_{T'}} w_i - w_j \tag{1}$$

Then, for each pair T, T' of topics, a link is created if and only if the result of the function $d(T, T')$ is below a threshold ϕ defined at the beginning of the analysis.

C. Extracting sentiment information focused on Trends Detection

Once a period is over, it's necessary to complement this factual information with sentiment information extracted from opinionated documents retrieved from social networks. Even though there are many sources for opinionated documents. The ones that reflect more clearly if a topic is being trendy or not are those present in social networks.

In order to extract opinions from such documents, an algorithm based on lexicon data is used matching the sentiment polarity or scores for each *positive*, *objectivity*, and *negative* terms. The usage of lexicons in opinion mining models is based on the hypothesis that a word can be considered as fundamental knowledge unit of an opinion, and therefore it can shed some lights on the sentiment polarity of a document as a whole.

In this research, the *SentiWordNet* [19] platform is used as a resource lexicon information, in which the labeled information is described as:

$$\vec{w} = \langle w, w^p, w^o \rangle \tag{2}$$

With $\vec{w}$ the labeled vector for the word w, w^p its positive sentiment score, w^n its negative sentiment score and w^o its objectivity score. Furthermore, every labeled word in *SentiWordNet* fulfills the equation 3:

$$w^p + w^n + w^o = 1 \tag{3}$$

Thus, given a set $\vec{w}_d$ of size k consisting of every word present in an opinionated document d, is possible to associate it to its corresponding sentiment scores as shown in 4:

$$d^p = \frac{\sum_{i=1}^{k} w^p}{\|\vec{w}_d\|}, d^n = \frac{\sum_{i=1}^{k} w^n}{\|\vec{w}_d\|}, d^o = \frac{\sum_{i=1}^{k} w^o}{\|\vec{w}_d\|} \tag{4}$$

Then, considering a method `polarity(document)` that given an opinionated document d returns its sentiment vector (d^p, d^n, d^o) and a set $\{d_i\}_{i=1...\mathbf{N}}$ of opinionated documents, the procedure shown in III.1 is performed to assign sentiment scores to each element of this set:

Algorithm III.1: Classification of opinionated documents

Data: $\{d_i\}_{i=1...\mathbf{N}}$
Result: $\{\vec{d_i}\}_{i\in\mathbf{N}}$

```
documents := [];
for i ← 1 to N do
    d_i ← polarity(d_i);
    documents.append(d_i);
return documents;
```

To determine which documents will be retrieved from the social networks being mined, a simple permutation is used to generate the queries. In this case, for a given topic T extracts all the *n-grams* of length n which characterize it during the period t.

Finally, for each topic T a set of queries is generated and a series of queries as performed on the mined social networks. In particular, our research will focus solely on Twitter as the social network to be mined.

D. Expanding the set of crawled feeds

Most of the retrieved documents in the crawling phase contain hyperlinks pointing to different websites that talk about the topics being discussed on them and, independently

of the nature of these hyperlinks, this new set of information allows the inclusion and evaluation of new feeds to our set of crawlable feeds.

Several approaches have been developed to allow the discovering of blog communities based on the relevance of the content published among a given set of blogs [20], [21]. Given that the presented methodology focuses on detecting trends in a given domain of knowledge, is expected that blogs discussing topics belonging to any given domain can be grouped in a blog community.

As such, we propose a methodology for expanding the set of feeds being mined that consists of two steps: the first step detects a set of potentially useful feeds based on how frequently are mentioned in the documents already retrieved, and then a second phase which evaluates each potential feed to see if they belong to a similar blog community and therefore their contents add valuable information to the topic mining algorithm.

The method `extractFeedURLs` extracts all urls of a document with some caveats: as these documents are published in blogs that are financed by advertising, many of these urls correspond to ads and they should be ignored as they will never provide useful information. Furthermore, taking the complete url, or just taking the domain is not enough as our objective is to detect potential additions to our feed set, so in order to avoid this a set of url stemming rules are defined:

- If the URL has a *query* component, it must be removed. The *query* component of a url is the one that comes after a question mark `?` and contains information to be sent to the server, such as marketing campaign information, search queries, etc.
- If the URL points directly to a file (e.g. `html`, `pdf`, `php`) only the domain name will be used.

Algorithm III.2: Detection of potential sources

```
Data: {d_i}_{i∈ℕ}
feasibleFeeds= [];
forall the document ∈ {d_i}_{i∈ℕ} do
    feeds = extractFeedURLs(document);
    forall the feed ∈ feeds do
        if database.updateFeedCount(feed) then
            feasibleFeeds.append(feed);
average =
database.getFeedCountAverage(feasibleFeeds);
forall the feed ∈ database.getFeedData(feasibleFeeds)
do
    if feed.count > average then
        createPossibleNewFeed(feed);
```

Only the amount of feeds that show a given url is considered in our proposed approach as it outperformed a frequency based. The reason behind this is the different styles of citation and hyperlink used by different blogs, if a frequency based approach was considered, the potential feed selection algorithm became biased towards the urls shown in those feeds who had a more aggresive citation style (i.e. they added a lot of hyperlinks to a document) than in those with a more passive citation style (i.e. those who add a couple of citations at the end of the document).

Function `database.updateFeedCount(feed)` increases by one as the source appears in the data, and returns `true` if the source has not been yet moved to the list of sources to evaluated, or `false` otherwise.

Function `getFeedCountAverage(feeds)` is in charge of getting the average between all the succesfull appearances of the input URLs. Then, function `createPossibleNewFeed(feed)` creates a new entry in the list of candidate sources to be evaluated in the second step of this methodology, and marks as processed the new source so it will be ignored in future iterations. This way, it will only consider the list of potential sources all the URLs which have a frequency higher than the average.

To evaluate these feasible feeds, a variation of the weblog communities discovery algorithm by Bulters et al.[20] focused in using topic information to create communities will be used.

Once a feed has been added to the feasible feed set, it starts to be crawled but the stored documents won't be used by any of the previously mentioned phases. Then, its *linkStrength* (step 2 of the methodology in [20]) shown in Eq. 5 is calculated between the feasible feed f and each one of the feeds f' being used to extract topic information. If the amount of feeds that have a *linkStrength* greater than σ (using a origin point the candidate feed) is greater than ρ, the feed will be added to the set of processed feeds.

$$\begin{aligned} linkStrength(f, f') \quad = \quad & w_{relev} \cdot relev \\ & + w_{reciprocity} \cdot recip \\ & + w_{cocitation} \cdot cocit \end{aligned} \tag{5}$$

The relevance, reciprocity, and cocitation terms are defined by Eq. 6, Eq. 7, and Eq. 8 respectively. A document d contains relevant content if it contains a certain percentage of the top N keywords of a topic t, for any element of the set of topics $\{t_i\}_{i\in\mathbb{N}}$ that belong to the documents retrieved from f'. Let r_d be 1 if a document d is relevant, 0 otherwise.

$$relev = \frac{\sum_{d \in D_f} r_d}{\|D\|} \tag{6}$$

$$recip = \begin{cases} 1.0 & \text{if } f'.linkSet \text{ has a link to } f \\ 0.0 & \text{Otherwise} \end{cases} \tag{7}$$

$$cocit = \frac{\|f.linkSet \cap f'.linkSet\|}{\|f.linkSet\|} \quad (8)$$

The weights used for calculating the *linkStrength* (Equation 5) are those approximated in [20], which are $w_{relev} = 0.5$, $w_{co-citation} = 0.3$, and $w_{reciprocity} = 0.2$.

This methodology it is described in III.3, which recieves as input parameter the potential source F_p to evaluate, and the threshold value ρ which will be used to decide whether F_p will be included into the set of analyzed sources.

Algorithm III.3: Evaluation of potential sources

Data: F_p, ρ
relatedFeeds = 0;
actualFeeds = database.getFeeds();
forall the $feed \in actualFeeds$ **do**
  if $linkStrength(F_p, feed) > \sigma$ **then**
    relatedFeeds++;
  if $\left(\frac{relatedFeeds}{actualFeeds.length}\right) > \rho$ **then**
    database.addFeed(F_p);

IV. Experimental Results

The described methodology was applied over a set of 20 feeds discussing technology topics over a period of 8 months. *RSS* (Real Simple Syndication) feeds were used because they show the most complete amount of metadata, and also because the way documents are presented in an *RSS* feed is easy to process and allows to passively poll for new documents without abusing the servers of our content providers.

For each retrieved document from the *RSS* feeds, the following information was stored: original content in *HTML* format, published date, original url, publishing feed, creation timestamp and any metadata contained within the *RSS* document.

A. Structure and Content Processing

Every document retrieved by the crawling processes is stored as raw data (i.e. with *HTML* tags, external links, navigation links, etc.) and prior to being used by both topic modeling and opinion mining algorithms they are pre-processed through the application of standard data cleaning methodologies such as the removal of *HTML* elements, extraction of *stop-words* and stemming. The crawler used for retrieving factual documents possess the capability of updating documents if they change after they were initially stored if these changes were explicitly registered by the feed being mined in order to obtain a more realistic representation of the source.

B. Feed set expansion Algorithm

The objective of this experiment is to measure the effectiveness of the relevance classification algorithm for feasible feeds. Thus, is needed to determine the existence of relationship between the feed being evaluated and the initial set of feeds. To assert if a relationship exists between them, a manual analysis of the topics discussed in each feed was performed.

1) Discovery of feasible feeds: To evaluate the algorithm for discovering feasible feeds, every feed present in a set of retrieved documents was manually classified as relevant or not relevant. The criteria used to define a feasible feed as relevant was if the content published by the feed pertains to the same area of knowledge as the feeds being mined, in the case of this experiment it would be if the documents discuss any kind of technology related events or entities.

To evaluate the algorithm that creates the set of feasible feeds, the following terms are defined:

1) $RPSS$ = Relevant potential sources selected to be evaluated.
2) PSE = Potential sources to be evaluated.
3) RPS = Relevant potential sources.

Then the precisicion and recall that evaluates the quality of Algorithm III.2,

$$Precision_{\text{Sources}} = \frac{RPSS}{PSE} \quad (9)$$

$$Recall_{\text{Sources}} = \frac{RPSS}{RPS} \quad (10)$$

2) Evaluation of feasible feeds: Once the set of feasible feeds was determined, the evaluation algorithm of potential feeds was ran after after two weeks of crawling each feasible feed discovered and that they were manually classified as relevant or not relevant. Finally, each potential feed was crawled during a period of two weeks and the relevance classification algorithm was applied to each one of them using as input data the documents retrieved during this period.

Let,

1) $RSUAC$ = Relevant sources under analysis classified as relevant.
2) SUA = Sources under analysis classified as relevant.
3) $RSUA$ = Relevant sources under analysis.

To evaluate the relevance classification algorithm the metrics shown in 9 and 10 where used.

$$Precision_{\text{Analysis}} = \frac{RSUAC}{SUA} \quad (11)$$

$$Recall_{\text{Analysis}} = \frac{RSUAC}{RSUA} \quad (12)$$

3) Experiment results: The results of the experiment for the discovery of feasible feeds are shown in the tables I and II. A total of 12000 documents were chosen and 31778 relevant links were extracted, from which 1493 correspond to unique feeds. These links were distributed as follows:

Type	Quantity
Feeds being mined	27,740
File	263
Social Networks (Facebook, Twitter, etc.)	397
Streaming Sites	45
Government Websites	122
Encyclopedic Websites (Wikipedia, IMDB, etc.)	234
Feed Aggregators	192
University Websites	33
Others	2,752

Table I
DISTRIBUTION OF LINKS BY TYPE

Links	Feeds
1	1,061
2	232
3 - 10	143
11 - 100	32
100 or more	5
Feeds being mined	20

Table II
AMOUNT OF FEEDS BY AMOUNT OF DOCUMENTS MENTIONING THEM

The average of document-citations for these links is 2.4, thus 180 feeds are included in the set of feasible feeds. Of these feasible feeds, 79 correspond to web sites of services, products or brands, and 101 to blogs, news sites or similar web sites where 61 of them published mainly technology related articles, and the rest of them published general interest news which included technology articles.

The $Recall_{\text{source}}$ of Algorithm III.3, using Eq. 10, is 0.35. Even though its recall is low because the amount of feasible feeds tend to increase as the evaluation period increases, as the majority of these feeds only appears in one or two documents, it can be considered that this low recall doesn't imply a loss of valuable information. In fact, if the $Recall_{\text{source}}$ is calculated not considering those feeds that show up only in one document, it goes up to 0.58. Furthermore, the $Precision_{\text{source}}$ using Eq 9 is 0.56 mainly due to the amount of web sites of services and products which many blogs in the technology area use to mention either because of a product launch or a review.

The experiment proposed to evaluate the relevance classification algorithm was ran with multiple values of ρ. Their precision (equation 11) and recall (equation 12) are shown in the table IV

In the table IV it can be observed that neither the recall or the precision of the algorithm could be calculated if a high enough value of ρ was used because no feed was classified as relevant. Furthermore, the *precision* of this algorithm increases with higher values of ρ due to the higher requirements for the feed to be more related with technology, and in the other hand, the *recall* decreases because the amount of selected feeds is lower due to the higher restrictions. To pick an optimal value for ρ, the one with the best precision possible must be used, because if a wrong feed is included, the higher the possibilities to introduce noise or wrong data in the models and therefore they will perform worst over time. Even if picking the best precision possible implies a lower recall as seen in the table IV, as long as relevant feeds are being included the algorithm is useful.

ρ	Precision	Recall
0.3	0.32	0.51
0.5	0.53	0.42
0.6	0.57	0.30
0.8	-	-

Table III
Precision AND *Recall* FOR MULTIPLE VALUES OF ρ

C. Model Validation and Trend Visualization

The purpose of evaluating this model is to determine its capability of representing how media reacts towards events that occur during the period where the analysis is being done. The events focused by this research will be called as *significant event* and are defined by:

Definition 2: **Significant Event**: If between two consecutive periods t_i and t_{i+1}, the difference between the amount of factual documents published $\frac{\|\vec{D}_{t_{i+1}}\| - \|\vec{D}_{t_i}\|}{\|\vec{D}_{t_i}\|}$ is greater than a threshold ρ, then a significant event occurred in t_{i+1}.

An example of a topic containing a significant event can be seen in 1, in which the sentiment associated with it its shown as a spline, and the amount of factual documents where the topic is mentioned. Given the big increase in factual documents between periods 5 and 6, a significant event is marked in period 6. Furthermore, it can be seen how the media coverage and the sentiment in social networks change over time.

To evaluate the proposed framework, the following approach was taken: for each topic, their corresponding time series will be manually analyzed for *significant events*, and the precision of the framework will be the precision of the algorithm regarding the amount of significant events, i.e. if a major event happened in the same period as a *significant event* that is shown by the methodology, then it's counted as a success. Let,

1) $SECC$ = Significant events correctly classified as such (manual annotation).
2) $ASEM$ = Amount of significant events found by the algorithm.

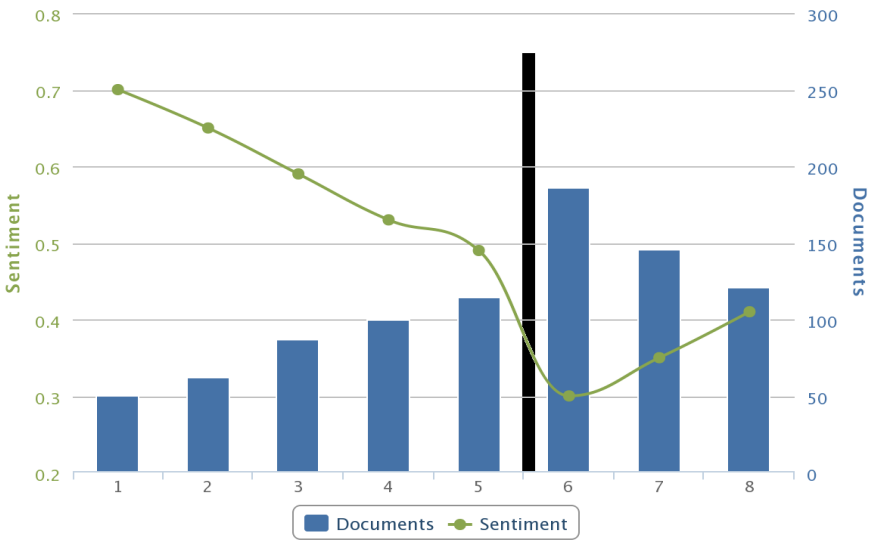

Figure 1. Evolution of a topic over time

3) $ASEF$ = Amount of significant events found manually.

Events that could be considered as "significant events" are manually annotated. The precision of the methodology is calculated by the Eq. 13, recall by Eq. 14, and F-measure by Eq. 15.

$$Precision_{SE} = \frac{SECC}{ASEM} \tag{13}$$

$$Recall_{SE} = \frac{SECC}{ASEF} \tag{14}$$

$$FMeasure_{SE} = 2 \cdot \frac{Precision_{SE} \cdot Recall_{SE}}{Precision_{SE} + Recall_{SE}} \tag{15}$$

1) Experimental Results: :

The proposed methodology was executed over a period of 8 months, during which a total of 200,890 factual documents were collected, out of which a 117 topics were extracted, and 268,800 tweets were retrieved. Also, a total of 65 *significant events* distributed over these topics were manually detected. To avoid getting incorrect results, only significant events that were detected after 6 documents were retrieved in an specific period of a topic were used for these calculations.

For value of ρ of 0.6 or higher, no significant events were found given that the amount of news on a weekly basis for technology topics can't match the amount of new news needed to be qualified as a significant event.

ρ	$Precision_{SE}$	$Recall_{SE}$	$FMeasure_{SE}$
0.2	0.25	0.71	0.37
0.3	0.38	0.65	0.48
0.4	0.48	0.57	0.52
0.5	0.61	0.51	0.56

Table IV
$Precision_{SE}$, $Recall_{SE}$, AND $FMeasure_{SE}$ FOR MULTIPLE VALUES OF ρ IN ALGORITHM III.3.

V. CONCLUSION

We conclude that the methodology presented in this paper is a feasible approach to model how trends could be represented on the Web as an interaction of events, topics, and the opinions expressed by their users on social networks.

This approach takes advantage of both factual and opinionated documents on the Web to create a visual representation of topics. This way, it allows the development of more advanced methodologies and frameworks focused on detection and modeling of trends on the Web through the extension to further application domains. For example, the inclusion of comments in news sites, which can lead to correlate how news describe a given event and the opinions expressed on the Web about it.

Given the broad definition of what a trend is and the even broader spectrum of variables that could be taken into consideration to detect them, it must be noted that this research proposes a basal approach towards this end. As such, this work is intended to be extensible and used as a framework from which several techniques could be

As future research branches we propose the inclusion of an algorithm with feature detection in the opinion mining phase. Also, improving the detection of significant events will allow the platform to better detect the appearance of trends over time. Furthermore, developing metrics of correlation between the information extracted from social media

and news source is desired. Also, the way of recovering factual or opinionated documents could be changed towards analyzing streams of data, allowing the development of a system capable of determining in advanced if significant events are going to happen, and if trends are being born.

ACKNOWLEDGMENT

This work was supported partially by the FONDEF project D10I-1198, entitled WHALE: *Web Hypermedia Analysis Latent Environment* and the Millennium Institute on Complex Engineering Systems (ICM: P-05-004-F, CONICYT: FBO16).

REFERENCES

[1] B. Ohana and B. Tierney, "Sentiment classification of reviews using sentiwordnet," *The Online Journal on Computer Science and Information Technology*, vol. 2, no. 1, pp. 120–123, 2009.

[2] S. Brody and N. Diakopoulos, "Cooooooooooooooollllllllllll-lll!!!!!!!!!!!!!! using word lengthening to detect sentiment in microblogs," in *EMNLP'11: Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*. Edinburgh, Scotland, UK.: Association for Computational Linguistics, 2011, pp. 562–570.

[3] A. Kao and S. Poteet, *Text mining and natural language processing*. Springer, 2007.

[4] E. M. Taylor, C. Rodriguez, J. D. Velásquez, G. Ghosh, and S. Banerjee, "Web opinion mining and sentiment analysis," in *Advanced Techniques in Web Intelligence-2*, J. D. Velásquez, V. Palade, and L. C. Jain, Eds. Springer, 2012, pp. 105–126.

[5] B. O'Connor, R. Balasubramanyan, B. Routledge, and N. Smith, "From tweets to polls: Linking text sentiment to public opinion time series," in *Proceedings of the International AAAI Conference on Weblogs and Social Media*, ser. ICWSM '10. AAAI Press, 2010, pp. 122–129.

[6] L. Sarmento, P. Carvalho, M. Silva, and E. de Oliveira, "Automatic creation of a reference corpus for political opinion mining in user-generated content," in *Proceedings of the 1st International CIKM Workshop on Topic-Sentiment Analysis for Mass Opinion*, ser. CIKM '09. Hong Kong, China: ACM, 2009, pp. 29–36.

[7] V. Sehgal and C. Song, "Sops: stock prediction using web sentiment," in *Data Mining Workshops, 2007. ICDM Workshops 2007. Seventh IEEE International Conference on*, ser. ICDMW '07. Omaha, Nebraska, USA: IEEE Computer Society, 2007, pp. 21–26.

[8] S. David and T. J. Pinch, "Six degrees of reputation: The use and abuse of online review and recommendation systems," *First Monday*, July 2006, special Issue on Commercial Applications of the Internet.

[9] A. Vakali, M. Giatsoglou, and S. Antaris, "Social networking trends and dynamics detection via a cloud-based framework design," in *Proceedings of the 21st international conference companion on World Wide Web*, ser. WWW '12 Companion. New York, NY, USA: ACM, 2012, pp. 1213–1220.

[10] R. Baeza-Yates, B. Ribeiro-Neto *et al.*, *Modern information retrieval*. ACM press New York., 1999, vol. 463.

[11] N. S. Glance, M. Hurst, and T. Tomokiyo, "Blogpulse: Automated trend discovery for weblogs," in *In WWW'04: Workshop on the Weblogging Ecosystem: Aggregation, Analysis, and Dynamics*. ACM, 2004.

[12] M. Mathioudakis and N. Koudas, "Twittermonitor: trend detection over the twitter stream," in *Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD '10. Indianapolis, Indiana, USA: ACM, 2010, pp. 1155–1158.

[13] I. P. Cvijikj and F. Michahelles, "Monitoring trends on facebook," in *Proceedings of the 2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing*, ser. DASC '11. Sydney, Australia: IEEE Computer Society, 2011, pp. 895–902.

[14] J. D. Velásquez, "Web site keywords: A methodology for improving gradually the web site text content," *Intelligent Data Analysis*, vol. 16, no. 2, pp. 327–348, 2012.

[15] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning*. The MIT Press, 2009.

[16] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, March 2003.

[17] B. Liu, "Opinion mining and summarization - sentiment analysis," 2008.

[18] E. Kouloumpis, T. Wilson, and J. Moore, "Twitter sentiment analysis: The good the bad and the omg!" in *ICWSM'11: Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*. AAAI, 2011, pp. 538–541.

[19] A. Esuli and F. Sebastiani, "SentiWordNet: A publicly available lexical resource for opinion mining," in *Proceedings of the Third International Conference on Language Resources and Evaluation*, ser. LREC '06. Genoa, Italy: European Language Resources Association (ELRA), 2006, pp. 417–422.

[20] J. Bulters and M. de Rijke, "Discovering weblog communities: A content- and topology-based approach," in *Proceedings of the International Conference on Weblogs and Social Media*, ser. ICWSM 07'. Boulder, Colorado, USA: AAAI, 2007, pp. 211–214.

[21] Y. Lin, H. Sundaram, Y. Chi, J. Tatemura, and B. Tseng, "Discovery of blog communities based on mutual awareness," in *Proceedings of the 3rd Annual Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, 15th World Wide Web Conference*, May 2006.

B. Listado de *stop-words* en español

a	a base de	a pesar de	a pesar de que
abajo	aca	acá	ademas
además	ahi	ahora	ahí
al	al contrario	al parecer	al respecto
al respecto	algun	alguna	algunas
alguno	algunos	algún	alla
alli	allá	allí	ambas
ambos	ante	anterior	antes
aparte	aquel	aquellas	aquello
aquellos	aqui	aquél	aquélla
aquéllas	aquéllos	aquí	arriba
asi	asimismo	así	atras
atrás	aun	aunque	aún
bajo	bastante	bien	bueno
buenos	cada	casi	caso
casos	cierta	ciertas	cierto
ciertos	como	con	con base en
con respecto a	conseguimos	conseguir	consigo
consigue	consiguen	consigues	consiguiente
cosa	cosas	creo	cual
cuales	cualquier	cualquiera	cuando
cuanta	cuantas	cuanto	cuantos
cuestion	cuestión	cuya	cuyas
cuyo	cuyos	cuál	cuáles
cuándo	cuánta	cuántas	cuánto
cuántos	cómo	dado que	de
de	de esa forma	de esta forma	de hecho
de igual forma	de la misma forma	de tal forma	decir
del	demás	dentro	desde
despues	después	dice	dicen
dicho	donde	dónde	e
el	ellas	ellos	en

en base a	en cambio	en conclusion	en conclusión
en concreto	en consecuencia	en definitiva	en efecto
en el fondo	en otras palabras	en particular	en primer lugar
en realidad	en relacion a	en relacion con	en relación a
en relación con	en resumen	en segundo lugar	en suma
en síntesis	en tercer lugar	encima	ende
entonces	entre	era	eramos
eran	eras	eres	es
esa	esas	ese	eso
esos	esta	estaba	estado
estais	estamos	estan	estas
este	esto	estos	estoy
está	están	fin	fue
fuera	fueron	fui	fuimos
ha	hace	haceis	hacemos
hacen	hacer	haces	hacia
hago	han	harta	hartas
harto	hartos	hasta	hay
i	igualmente	incluso	ir
la	largo	las	le
les	lo	los	luego
manera	mas	me	mediante
mejor	mi	mientras	mio
mis	misma	mismas	mismo
mismos	modo	mucha	muchas
mucho	muchos	muy	más
mí	mío	ni	ningun
ninguna	ningunas	ningunos	ningún
no	no obstante	nos	nosotros
o	o sea	obvia	obvias
obvio	obvios	ojala	ojalá
ojalá	otra	otras	otro
para	pero	pese a	poca

pocas	poco	pocos	podeis
podemos	poder	podria	podriais
podriamos	podrian	podrias	podría
podríamos	podrían	podrías	por
por conclusion	por conclusión	por dicha razon	por dicha razón
por dicho motivo	por ejemplo	por el contrario	por esa razon
por esa razón	por ese motivo	por otra parte	por otro
por otro lado	por que	por qué	por su parte
por supuesto	por tal motivo	por tal razón	por un lado
por una parte	por último	porque	porqué
posteriormente	primero	puede	pueden
puedo	pues	puesto	que
que	quien	quiza	quizas
quizá	quizás	quién	qué
respecto a	respecto de	sabe	sabeis
sabemos	saben	saber	sabes
se	segun	segundo	según
según parece	sentido	ser	si
siendo	siendo	similar	sin
sin duda	sin embargo	sin lugar a dudas	sobre
sobre la base de	sois	sola	solamente
solas	solo	solos	somos
son	soy	su	super
sus	sí	sólo	súper
tal	tal vez	tales	también
tan	tanta	tantas	tanto
tantos	te	teneis	tenemos
tener	tengo	tercer	tercero
ti	tiempo	tiene	tienen
toda	todas	todo	todos
tras	través	tus	tuyo
u	ultimo	un	una
unas	uno	unos	usar

ustedes	va	vais	vamos
van	vaya	vemos	ven
ver	vez	visto	vosotras
vosotros	voy	vía	y
ya	yo	él	éramos
ésa	ése	ésos	ésta
éstas	éste	éstos	último

Cuadro 7.1: Listado de *stop-words* en español

C. Listado de *stop-words* en inglés

’ll	’ve	a	able	about
above	abst	accordance	according	accordingly
across	act	actually	added	adj
adopted	affected	affecting	affects	after
afterwards	again	against	ah	all
almost	alone	along	already	also
although	always	am	among	amongst
an	and	announce	another	any
anybody	anyhow	anymore	anyone	anything
anyway	anyways	anywhere	apparently	approximately
are	aren	arent	arise	around
as	aside	ask	asking	at
auth	available	away	awfully	b
back	be	became	because	become
becomes	becoming	been	before	beforehand
begin	beginning	beginnings	begins	behind
being	believe	below	beside	besides
between	beyond	biol	both	brief
briefly	but	by	c	ca
came	can	can’t	cannot	cause
causes	certain	certainly	co	com
come	comes	contain	containing	contains
could	couldnt	d	date	did

didn't	different	do	does	doesn't
doing	don't	done	down	downwards
due	during	e	each	ed
edu	effect	eg	eight	eighty
either	else	elsewhere	end	ending
enough	especially	et	et-al	etc
even	ever	every	everybody	everyone
everything	everywhere	ex	except	f
far	few	ff	fifth	first
five	fix	followed	following	follows
for	former	formerly	forth	found
four	from	further	furthermore	g
gave	get	gets	getting	give
given	gives	giving	go	goes
gone	got	gotten	h	had
happens	hardly	has	hasn't	have
haven't	having	he	hed	hence
her	here	hereafter	hereby	herein
heres	hereupon	hers	herself	hes
hi	hid	him	himself	his
hither	home	how	howbeit	however
hundred	i	i'll	i've	id
ie	if	im	immediate	immediately
importance	important	in	inc	indeed
index	information	instead	into	invention
inward	is	isn't	it	it'll
itd	its	itself	j	just
k	keep	keeps	kept	keys
kg	km	know	known	knows
l	largely	last	lately	later
latter	latterly	least	less	lest
let	lets	like	liked	likely
line	little	look	looking	looks

ltd	m	made	mainly	make
makes	many	may	maybe	me
mean	means	meantime	meanwhile	merely
mg	might	million	miss	ml
more	moreover	most	mostly	mr
mrs	much	mug	must	my
myself	n	na	name	namely
nay	nd	near	nearly	necessarily
necessary	need	needs	neither	never
nevertheless	new	next	nine	ninety
no	nobody	non	none	nonetheless
noone	nor	normally	nos	not
noted	nothing	now	nowhere	o
obtain	obtained	obviously	of	off
often	oh	ok	okay	old
omitted	on	once	one	ones
only	onto	or	ord	other
others	otherwise	ought	our	ours
ourselves	out	outside	over	overall
owing	own	p	page	pages
part	particular	particularly	past	per
perhaps	placed	please	plus	poorly
possible	possibly	potentially	pp	predominantly
present	previously	primarily	probably	promptly
proud	provides	put	q	que
quickly	quite	qv	r	ran
rather	rd	re	readily	really
recent	recently	ref	refs	regarding
regardless	regards	related	relatively	research
respectively	resulted	resulting	results	right
run	s	said	same	saw
say	saying	says	sec	section
see	seeing	seem	seemed	seeming

seems	seen	self	selves	sent
seven	several	shall	she	she'll
shed	shes	should	shouldn't	show
showed	shown	showns	shows	significant
significantly	similar	similarly	since	six
slightly	so	some	somebody	somehow
someone	somethan	something	sometime	sometimes
somewhat	somewhere	soon	sorry	specifically
specified	specify	specifying	state	states
still	stop	strongly	sub	substantially
successfully	such	sufficiently	suggest	sup
sure	t	take	taken	taking
tell	tends	th	than	thank
thanks	thanx	that	that'll	that've
thats	the	their	theirs	them
themselves	then	thence	there	there'll
there've	thereafter	thereby	thered	therefore
therein	thereof	therere	theres	thereto
thereupon	these	they	they'll	they've
theyd	theyre	think	this	those
thou	though	thoughh	thousand	throug
through	throughout	thru	thus	til
tip	to	together	too	took
toward	towards	tried	tries	truly
try	trying	ts	twice	two
u	un	under	unfortunately	unless
unlike	unlikely	until	unto	up
upon	ups	us	use	used
useful	usefully	usefulness	uses	using
usually	v	value	various	very
via	viz	vol	vols	vs
w	want	wants	was	wasn't
way	we	we'll	we've	wed

welcome	went	were	weren't	what
what'll	whatever	whats	when	whence
whenever	where	whereafter	whereas	whereby
wherein	wheres	whereupon	wherever	whether
which	while	whim	whither	who
who'll	whod	whoever	whole	whom
whomever	whos	whose	why	widely
willing	wish	with	within	without
won't	words	world	would	wouldn't
www	x	y	yes	yet
you	you'll	you've	youd	your
youre	yours	yourself	yourselves	z
zero	&			

Cuadro 7.2: Listado de *stop-words* en inglés