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

DISEÑO E INTEGRACIÓN DE UN MÓDULO PARA DETECTAR Y CATEGORIZAR OPINIONES DE RECLAMO EN UN SISTEMA DE ANÁLISIS WEB APLICADO AL RUBRO DE LAS TELECOMUNICACIONES

MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL INDUSTRIAL

JOAQUÍN ESTEBAN AGUILAR RUIZ

PROFESOR GUÍA:
JUAN DOMINGO VELÁSQUEZ SILVA

MIEMBROS DE LA COMISIÓN:
IGNACIO ALEJANDRO CALISTO LEIVA
ROCIO BELÉN RUIZ MORENO

Este trabajo ha sido parcialmente financiado por el proyecto CORFO 13IDL2-23170

SANTIAGO DE CHILE
2017

RESUMEN DE LA MEMORIA PARA OPTAR AL
TITULO DE: Ingeniero Civil Industrial
POR: Joaquín Esteban Aguilar Ruiz
FECHA: 03/04/2017
PROFESOR GUÍA: Juan Domingo Velásquez Silva

DISEÑO E INTEGRACIÓN DE UN MÓDULO PARA DETECTAR Y CATEGORIZAR OPINIONES DE RECLAMO EN UN SISTEMA DE ANÁLISIS WEB APLICADO AL RUBRO DE LAS TELECOMUNICACIONES

El objetivo general de esta memoria de titulo es diseñar e integrar un módulo para detectar y categorizar opiniones de reclamos en un sistema de análisis web utilizando herramientas de Machine Learning. Este trabajo se desarrolla dentro del marco del proyecto OpinionZoom, que es un proyecto de investigación y desarrollo concursado por InnovaChile de CORFO y dirigido por el Web Intelligence Centre (WIC) de la Universidad de Chile. Este proyecto busca aumentar el conocimiento que tienen las organizaciones sobre individuos pertenecientes a la industria que sirven, utilizando los datos públicos de los usuarios chilenos de Twitter.

En esta etapa de OpinionZoom surge la oportunidad de utilizar el contenido que los usuarios envían a las cuentas de Twitter de las empresas en referencia a los reclamos, cuyo interés nace de las entrevistas y requerimientos de los comienzos de este proyecto, en conjunto con el poco provecho que extraen las empresas de esta red social.

Esta memoria de título pretende comprobar la hipótesis de investigación de utilizar algoritmos de Data Mining y Machine Learning que permitan identificar opiniones de reclamos en Twitter y clasificarlas en categorías predefinidas. Lo que se divide en dos etapas, la primera consiste en identificar opiniones de reclamos en Twitter, algo que no ha sido investigado hasta el momento, y como segunda etapa categorizar el contenido de estos mensajes en función de los productos y servicios de las empresas. Para la validación de esta hipótesis se decidió por utilizar el segmento de mayor relevancia en términos de reclamos en la red social Twitter: Telecomunicaciones.

El potencial de lo desarrollado en este trabajo es entregar información útil a la empresas sobre los reclamos, a nivel agregado, que sus clientes y usuarios expresan en la red social de Twitter, de modo de que puedan tomar mejores decisiones.

Se utilizó un set de datos que abarcó toda la historia de cuatro cuentas de Twitter, cuya elección se basó en la relevancia en el ámbito de los reclamos en la red social. Los datos se modelaron con el enfoque Bag-Of-Words y se implementaron 4 algoritmos de Machine Learning para clasificar los tweets en una primera etapa dentro de 4 clases, incluida reclamo, y en una segunda etapa en 9 categorías predefinidas. Para la primera etapa el algoritmo Support Vector Machines entregó los mejores resultados con un *F-Measure* de 0.823 para la clase Reclamo. Y para la segunda etapa los mejores resultados se lograron en Support Vector Machines y Decision Trees con un *accuracy* de 81.3 %. Lo que permite validar la hipótesis de investigación. Finalmente, se diseñó e implemento el módulo para detectar y categorizar reclamos en la plataforma web de OpinionZoom.

A Roger

Agradecimientos

Quisiera empezar por agradecer a mis padres, Carmen y Luis, por haberme dado todo en la vida y que nunca me haya faltado nada. Sin ellos esto no habría sido posible. Siempre han estado cuando los he necesitado. También agradecer a mi hermana con quien viví y compartí toda mi vida universitaria.

Agradecer también a todos los amigos que conocí en la universidad y que estuvieron siempre presentes en todos estos años: Felipe, Pedro, Ismael, Olate y Tchimino.

Este proceso habría sido imposible sin mi deporte favorito, el tenis, sin duda ha sido uno de mis motores en toda mi vida universitaria, quisiera agradecer a Rodolfo por todos los momentos felices que disfruté realizando este deporte. Gracias también a mi partner Lorca.

Finalmente, agradecer al profesor Juan Velasquéz, por la oportunidad de realizar este trabajo y a todo el equipo del WIC que me ayudó en este proceso, en especial a Pangui, Seba e Ignacio.

Tabla de Contenido

Índice de tablas

Índice de figuras

Capítulo 1

Introducción

El acceso a internet en el mundo es cada vez mayor, se estima que existen 3.696.238.430 usuarios actualmente [1], cifra que el Chile abarca cerca del 70 % de la población al año 2015 [2]. Esta gran cantidad de usuarios ha potenciado la web, donde el contenido ya no es generado por las empresas y dueños de los sitios, por el contrario, la mayor cantidad de los datos son generados por los mismos usuarios, que son una parte activa de esta nueva era. Las redes sociales son la máxima expresión de este fenómeno, en ellas prácticamente el 100 % del contenido es producido por los usuarios.

Chile es uno de los países con la mayor penetración de redes sociales en el mundo. Facebook es la red social con más usuarios en nuestro país, alcanzando un 88 % de la población [3], seguido de Youtube con un 85 %, y más abajo en la cuenta se encuentra Twitter con un 47 % de usuarios. Se estima que en Chile existen cerca de 2.000.000 de usuarios.

Actualmente en Chile existen más de 5.000 organizaciones que cuentan con perfiles en redes sociales y la gran parte de ellas desconocen el verdadero potencial que se puede obtener de los datos públicos que los usuarios emiten día a día. Más del 30 % de las empresas que poseen presencia en redes sociales no poseen una estrategia clara, no realizan acción alguna con la información que sus usuarios depositan en sus cuentas en las redes sociales [4]. La otra parte de las empresas posee una estrategia definida, sin embargo, su forma de acción es más operativa que gerencial, es decir, no toman decisiones en base a la información y el conocimiento que se pueda extraer de las redes sociales, tan solo se limitan a responder mensajes individualmente.

Los principales usos que le dan las empresas a las redes sociales es como un medio para publicitar servicios, productos, concursos, responder reclamos y consultas de forma particular, pero no hacen un análisis general de la información que les llega de parte de sus clientes y potenciales clientes. De este modo surge la oportunidad de analizar los comentarios públicos que los usuarios emiten a las empresas en la red social de Twitter en relación a los reclamos, de modo de entregar información útil para que puedan tomar mejores decisiones.

1.1. Antecedentes

Este trabajo se enmarca en el Proyecto OpinionZoom, financiado por INNOVA CORFO, el cual se realiza en el Centro de Inteligencia Web de la Universidad de Chile.

1.1.1. Web Intelligence Centre - WIC

El Web Intelligence Centre [5] (WIC) es el centro de inteligencia Web de Ingeniería Industrial de la Universidad de Chile, tiene alrededor de una década de antigüedad y ha estado a cargo durante toda su historia por el profesor Juan Velásquez. Este centro cuenta con numerosas publicaciones en revistas científicas internacionales, posee varias memoria de pregrado y tesis de postgrado, y año a año recibe aportes de profesionales y estudiantes que llegan al centro. En su pagina web el WIC declara como misión, visión y objetivos:

Misión - ¡Creando soluciones ingenieriles inteligentes!

Desarrollar investigación de frontera en el campo de Tecnologías de información creando nuevas soluciones para abordar problemas complejos de ingeniería utilizando herramientas basadas en la Web de las Cosas.

Visión - Impactar en el mundo

Ser un líder a nivel internacional en la investigación de tecnologías de información y comunicaciones aplicadas a la resolución de problemas del mundo real.

Objetivos

- Publicar en las principales revistas, conferencias y editoriales relacionadas con Web Intelligence.
- Proveer a servicio profesional , excelente y rápido para todos nuestros clientes.
- Dictar cursos de orientación practica acerca de las tecnologías de información y su aplicación en los negocios.

El WIC dicta el contexto de investigación y desarrollo en que enmarcan proyectos como: OpinionZoom, AKORI, KOKORO, DOCODE y WHALE.

1.1.2. Proyecto OpinionZoom

OpinionZoom [6] es un proyecto de Web Opinion Mining del WIC que utiliza los datos depositados en Twitter: los procesa, analiza, y entrega información útil a las empresas e instituciones para conocer mejor a sus clientes y mercado objetivo.

1.1.2.1. Objetivos de OpinionZoom

Objetivo general

Aumentar el conocimiento que tienen las organizaciones sobre individuos pertenecientes a la industria a la que sirven, utilizando para ello los datos públicos de los usuarios chilenos de Twitter.

Objetivos Específicos

1. Construir un sistema que recopile los tweets, los perfiles y las conexiones de los usuarios chilenos de Twitter, los almacene y entregue información obtenida del procesamiento de ellos a través de una plataforma Web.

2. Diseñar un plan de negocios para capitalizar el valor que se puede crear con esta plataforma.

3. Comercializar el uso de la plataforma hasta conseguir que su mantención sea sustentable.

1.1.2.2. Visión, Misión y Valores de OpinionZoom

La estrategia del proyecto se compone de una misión, una visión, valores y objetivos [7, 8].

Visión

"Ser el observatorio de Redes Sociales más confiable y cercano de Chile, siendo para las organizaciones un socio clave para entender lo que ocurre en el mundo de habla hispana desde una óptica de negocios"

Misión

"Crear valor para las industrias de servicios en Chile, ayudándoles a tomar mejores decisiones a través de la minería de datos en las redes sociales"

Valores

Todo el proyecto se enmarca en valores que orientan y canalizan las acciones de OpinionZoom:

- Respeto por la privacidad de los datos de los usuarios. OpinionZoom no identifica a personas naturales con cuentas de Twitter, por lo que sólo se ocupan datos públicos para su caracterización.
- Ausencia de datos sensibles en los análisis del sistema. OpinionZoom no promueve ni concibe la investigación dirigida a encontrar orientación sexual, preferencia política, origen racial u opción religiosa de las personas.
- Innovación permanente para responder a los desafíos dinámicos en el mundo de la tecnología, buscando estar siempre en el estado del arte a nivel mundial. La investigación es el sello del proyecto y de la Universidad que lo alberga, por lo que dejar de innovar no se puede permitir hasta el fin del proyecto.
- Orientación al cliente. Pese a anticipar sus necesidades, la idea es trabajar juntos para que

las soluciones propuestas respondan efectivamente a sus problemas reales. Por lo mismo OpinionZoom tomará como algo prioritario los requerimientos y sugerencias de sus clientes y procurará prestarles una atención personalizada que asegure la calidad de los servicios entregados.

1.1.2.3. Funciones y Servicios de OpinionZoom

En un comienzo el proyecto OpinionZoom contaba con tres secciones derivadas de la elaboración del modelo de negocios [7], las cuales fueron refinadas en [8, 9], focalizándose en 4 funcionalidades:

1. **Artículos de Interés General:** Entrega artículos de interés general obtenidos de la minería de opiniones en Twitter. A estos artículos la comunidad puede acceder de forma gratuita del Home de la pagina Web de OpinionZoom, y tienen como principal objetivo atraer la atención de posibles clientes mostrándoles las capacidades del sistema.

2. **Buscador de Tweets:** Tiene como finalidad explorar preliminarmente lo que los usuarios de la red social Twitter hablan sobre un tema o keyword[1]. En complemento se agregan dos columnas para enriquecer el contenido, la primera indica la polaridad del tweet y la segunda entrega el impacto[2] del tweet.

3. **Inteligencia de Clientes:** Permite capturar información de los usuarios de las redes sociales, a partir de sus comentarios o perfiles, etiquetándolos y clusterizándolos según sean las necesidades del cliente. Se busca analizar información en torno a una o varias palabras claves (keywords) que sean de interés del cliente, desprendiéndose de esa forma datos como la polaridad de la keyword, frecuencia, impacto, etc.

4. **Sistema de Alertas:** Permite generar alertas al cliente en función a reglas de negocio que imponga (cierto nivel de polaridad, influencia, impacto, entre otro), con el propósito de monitorear en tiempo real los temas de interés que pueda tener.

Actualmente el proyecto OpinionZoom se encuentra en continuo proceso de validación con potenciales clientes. Se está realizando una segunda iteración del modelo de negocios propuesto en [7] con el objetivo de redefinir los servicios que ofrece la aplicación y adecuarlos a la demanda existente en el mercado. Se está adaptando el sitio web de OpinionZoom para mejorar la usabilidad y utilidad de las funciones que se ofrecen.

Además, los algoritmos base del sistema se encuentran en proceso de mejora continua, en esta etapa del proyecto se están mejorando los siguientes:

- Polaridad
- Detección de genero
- Calculo de edad
- Calculo de influencia

[1] palabras claves (con menos de 50 caracteres)

[2] indicador que se calcula como la polaridad de una opinión ponderada por la influencia que le corresponde.

1.2. Descripción y Justificación

La información obtenida a partir del contenido generado por usuarios en la web puede tener muchos usos. Uno de ellos es identificar cuando los usuarios se expresan con disconformidad hacia una empresa, en otras palabras corresponde a determinar opiniones de reclamo o disgusto frente a una empresa por sus productos y/o servicios. En la red social de Twitter se deposita constantemente información que los usuarios emiten y las empresas no realizan gestión a nivel general de estos datos, tan solo realizan acciones particulares al responder uno a uno los mensajes [4]. Por esto surge la oportunidad de contar con esta información de manera automática, la cual puede ayudar a las organizaciones a detectar cuales servicios presentan problemas para sus consumidores, de modo de poder generar acciones y tomar mejores decisiones frente a esto.

Otro uso interesante que se le puede dar al contenido generado por usuarios en la web es la clasificación en torno a un tema o término en común. Dentro de la red social Twitter se puede segmentar el contenido por medio de *#Hashtags*, los cuales corresponden a términos o conjuntos de palabras que permiten la indexación sobre un tema y se encuentran dentro del texto de un tweet. Esta segmentación que facilita Twitter queda limitada al momento de querer agrupar el contenido referente a un tema común en donde no exista la presencia de *hashtags*. Además hay que considerar que para referirse a un tema existen múltiples formas para hacerlo, por lo que es necesario identificar términos semejantes (sinónimos) con el fin de poder agrupar el contenido similar.

Dado el potencial que se tiene al identificar opiniones de reclamos y el poder agruparlos dentro de un tópico, ha surgido un área de investigación llamada *Topic Classification*, donde se utilizan técnicas de análisis de datos, como Opinion Mining, Text Mining y Machine Learning para clasificar texto en diferentes clases. Actualmente no existe ninguna investigación en donde se busque analizar las opiniones de reclamos en Twitter, lo más cercano que se ha realizado en este ámbito es lo realizado en [10], donde los autores diseñaron un clasificador capaz de identificar texto con carácter de reclamo. Sin embargo, existen grandes diferencias entre esta investigación con lo desarrollado en este trabajo de título. En primer lugar, el dominio desde donde proviene el texto generado en dicha investigación posee un lenguaje más rico en contexto y contenido que el que se encuentra en Twitter, en segundo lugar el idioma es distinto, y el tercer lugar, y más importante a tener en cuenta, es la extensión del texto, los tweets están acotados a un máximo de 140 caracteres, lo que genera una mayor dificultad al momento de clasificar un texto, dada la falta de contexto.

Por otro lado, la oportunidad de este trabajo de título surge en las diversas entrevistas realizadas en [7], en donde las organizaciones expresan interés por la información que se pueda extraer en relación de los reclamos acerca de sus productos y/o servicios desde las redes sociales. En mayor detalle dentro de los intereses mostrados por las organizaciones entrevistadas se tiene:

1. **Segmentación:** Realizar una clasificación de los reclamos en distintas categorías ad hoc con los productos y servicios que la empresa ofrece, con el objetivo de ayudar a las gestión.

2. **Indicadores y Métricas:** Generar métricas e indicadores en relación a los reclamos: frecuencia de reclamos, cantidad de reclamos por categoría, cantidad de reclamos resueltos, cantidad de reclamos pendientes, etc.

3. **Tiempo de respuesta:** Guardar los tiempos en que se genera el reclamo y los tiempo en que

la empresa responde, de modo de analizar las diferencias de tiempo entre que se genera un reclamo y se logra solucionar.

4. **Datos de Usuario:** Integrar datos de usuario con el fin de analizar su historial a través del tiempo con la empresa. Visualizar reclamos anteriores, influencia, etc.

5. **Identificar Cliente:** Determinar si quien emite el reclamo es realmente un cliente de la empresa analizada de modo de no malgastar recursos resolviendo reclamos de quienes no son verdaderos clientes. O simplemente priorizar los reclamos que provienen de clientes reales.

6. **Otras Fuentes de Opinión:** Incorporar otras fuentes de opinión y redes sociales donde los usuarios y clientes exponen reclamos, tales como: Facebook, SERNAC, paginas Web de reclamos, etc.

Este trabajo de título tan solo se hace cargo de los dos primeros intereses mostrados por las empresas en las entrevistas realizadas. No es posible abarcarlos todos dado el alcance de esta memoria. Se pretende dar el inicio en el campo de los reclamos al diseñar y construir un clasificador que entregue información de utilidad a los clientes, por lo que los otros intereses manifestados en estas entrevistas quedan para una etapa posterior y trabajo a futuro a realizar.

El trabajo a desarrollar corresponde a un complemento para el proyecto OpinionZoom, el cual consiste en desarrollar un "Módulo de Reclamos", que tiene por objetivo el entregar información valiosa en relación al comportamiento de los reclamos en la red social de Twitter para que las empresas puedan tomar mejores decisiones. Más en específico, consiste en el diseño de una metodología para la creación de un algoritmo que permita identificar opiniones de reclamos dirigidas a las cuentas de Twitter de las organizaciones. Este algoritmo permitirá en una primera instancia detectar los comentarios que representan una opinión de reclamo y en una segunda instancia los categorizará en clases predefinidas con referencia al rubro o industria en la cual la empresa se desenvuelve. El desarrollo de este algoritmo contempla dos clasificadores, el primero detecta las opiniones de reclamos y el segundo las enmarca dentro de una clase que hace alusión a alguno de los productos y/o servicios presentes en el rubro al cual pertenezca la empresa.

Como segunda parte, luego de construido el clasificador, este trabajo de título contempla el diseño y posterior integración de un Módulo de Reclamos en la aplicación web de OpinionZoom. El objetivo de este es visualizar la información referente a reclamos, tal como: frecuencia de reclamos a través del tiempo en las cuentas de Twitter de las empresas, desagregado por categorías referentes a los servicios que ellas prestan a sus clientes.

1.3. Objetivos

1.3.1. Objetivo General

Diseñar e Integrar el módulo para detectar y categorizar opiniones de reclamos, en un sistema de análisis web utilizando herramientas de Machine Learning.

1.3.2. Objetivos Específicos

1. Evaluar el estado del arte referente a *Topic Classification*.
2. Construir un set de datos de tweets etiquetados que sirva como Data de entrenamiento.
3. Evaluar Algoritmos de clasificación que permitan identificar opiniones de reclamos, y clasificarlas en categorías predefinidas.
4. Diseñar e integrar el módulo funcional en la plataforma web de OpinionZoom.

1.4. Hipótesis de investigación

La hipótesis de investigación que se pretende validar y que orienta el presente trabajo de título es la siguiente:

"Es posible utilizar algoritmos de Data Mining y Machine Learning que permitan identificar opiniones de reclamos en Twitter y clasificarlas en categorías predefinidas."

1.5. Alcances

Este trabajo de título tiene el principal objetivo de verificar la hipótesis de investigación propuesta, por ende todo lo desarrollado se encuentra en un escenario muy controlado. Se utiliza un segmento muy acotado el cual no es aplicable directamente en el general de las cuentas en Twitter.

Dentro de las características de lo realizado se encuentra que:

- Los tweets considerados son solo los escritos en idioma español. Serán considerados solo los mensajes dirigidos a las cuentas de las empresas con presencia en la red social de Twitter.
- Se utiliza el rubro de las telecomunicaciones, considerado como el más apto para validar la hipótesis. Además dentro de este rubro, se selecciona un subconjunto de empresas (cuentas de Twitter) para evaluar la aplicabilidad de un modelo clasificador de opiniones de reclamos. Por ende, lo desarrollado no es aplicable directamente en otras empresas o rubro sin antes evaluar su desempeño, punto el cual esta memoria no se hace cargo.
- Este trabajo se hace cargo tan solo de los 2 primeros intereses mostrados por las empresas en [7]. Para los otros 4 intereses se debe tener construido un algoritmo que permita identificar opiniones de reclamos, por lo que se quedan fuera en este trabajo. Estos se dejan propuestos como trabajo futuro a realizar para mejorar el Módulo de Reclamos y dar una mejor propuesta de valor a los clientes de OpinionZoom.

por otro lado, este trabajo de título corresponde a un prototipo funcional y contempla una parte del proyecto OpinionZoom, por lo que quedan fuera otras etapas que ya fueron, o serán realizadas, en específico estas son:

- Evaluación económica del proyecto.
- Búsqueda de un cliente real.
- Diseño y creación de la aplicación web OpinionZoom.
- Test de usabilidad de la pagina Web.
- Algoritmo de polaridad de tweets en idioma español chileno.
- Algoritmo para detectar ironía en el texto.

1.6. Resultados Esperados

Los resultados esperados están en directa relación con los objetivos específicos descritos. Estos son:

1. Marco conceptual y Análisis las más recientes investigaciones realizadas en *Topic Classification*, con el fin de determinar las mejores prácticas y herramientas a utilizar.

2. Set de datos (tweets) etiquetados que permitan entrenar el algoritmo de Machine Learning.

3. Rendimiento de los algoritmos Analizados. Corresponde a los resultados de los diferentes algoritmos desarrollados, en otras palabras la matriz de confusión de cada modelo y sus respectivos indicadores de rendimiento.

4. Un modelo (algoritmo) que permita identificar opiniones de reclamos.

5. Un modelo (algoritmo) que permita clasificar opiniones de reclamos en categorías predefinidas en relación a productos y/o servicios que las empresas ofrecen.

6. Módulo funcional implementado dentro de la aplicación web de OpinionZoom con sus respectivas funcionalidades.

1.7. Metodología

El desarrollo de esta memoria tiene como punto inicial el investigar y analizar la literatura referente a los diferentes temas que enmarcan este trabajo. Se busca analizar las herramientas de Text Mining, Opinion Mining y Machine Learning, además, se busca estudiar literatura sobre Reclamos, y el impacto que genera en las empresas. Esto con el objetivo de comprender de forma íntegra el contexto en donde se enmarca el trabajo de título y poder abordarlo con mejores herramientas y conocimientos.

Como segundo para se tiene el analizar el estado del arte referente a clasificación automática de texto utilizando técnicas de Machine Learning en el dominio de Twitter. Aquí también se busca identificar los desafíos y las mejores prácticas para tratar con texto proveniente de esta red social.

Teniendo identificadas las herramientas y mejores prácticas a utilizar, la siguiente etapa es la

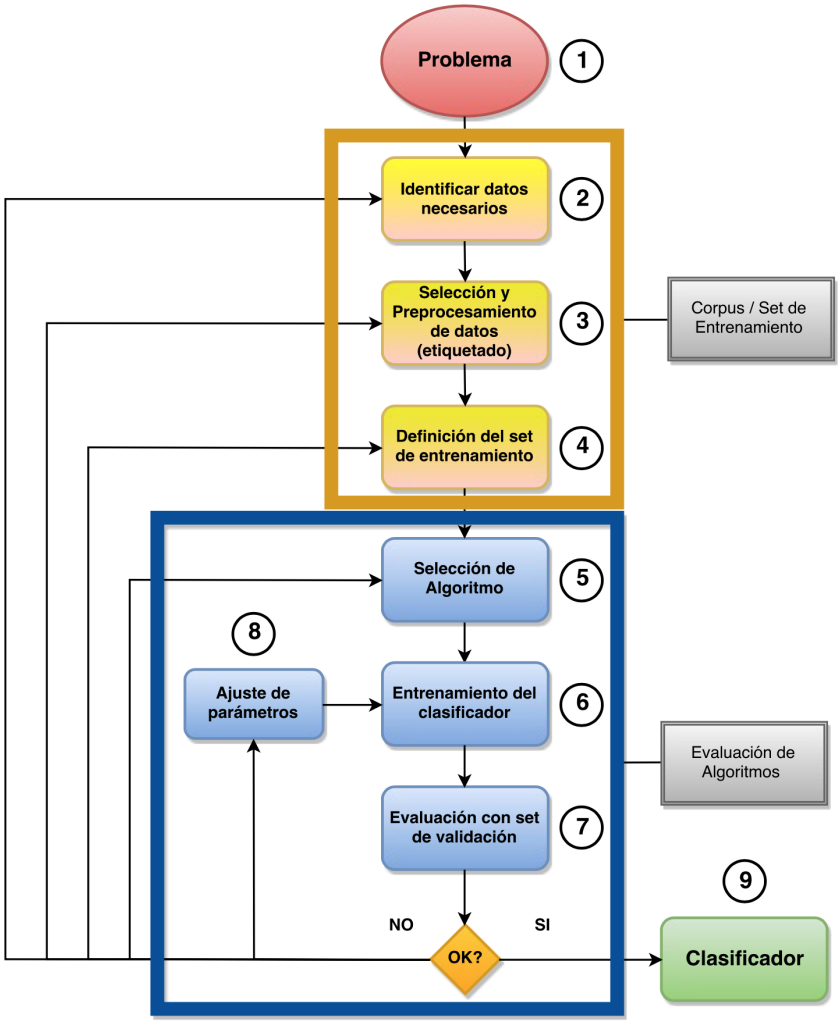

Figura 1.1: Metodología de un proceso de Machine Learning Supervisado.
Fuente: Imagen adaptada de [11].

construcción del algoritmo de clasificación y categorización de reclamos. Para lo cual se utilizará la metodología mostrada en la figura 1.1, la cual representa los pasos a seguir para el desarrollo de un algoritmo de aprendizaje supervisado y que consta de 9 etapas. A grandes rasgos esta metodología cuenta con dos macro-procesos: (1) diseño y construcción de un Set de datos / Set de entrenamiento, y (2) evaluación de algoritmos de clasificación supervisados.

En el primer macro-proceso busca la construcción del set de datos, lo que implica definir que datos se utilizarán, de donde se obtendrán, escoger los más representativos y etiquetarlos.

El segundo proceso busca evaluar distintos algoritmos de clasificación con el objetivo de encontrar el que mejor desempeño posea dado el problema a resolver. Para el desarrollo de esta memoria los algoritmos deben ser capaces de clasificar el texto en más de dos clases, lo que se denomina algoritmos multi-clase. Una vez evaluados todos los algoritmos a analizar se escoge el que mejores métricas e indicadores presente. Cabe destacar que esta metodología considera múltiples iteraciones, en donde se busca desde variar los parámetros del algoritmo hasta extraer nuevos datos.

Habiendo desarrollado el algoritmo de clasificación, se tiene como siguiente etapa el diseñar e integrar un módulo de reclamos funcional en la aplicación web de OpinionZoom. Esto conlleva el definir los componentes que debe incluir (gráficos, tablas, botones, etc.), así también como las métricas e indicadores a mostrar (por ejemplo: frecuencia por categorías, frecuencia de reclamos por mes, entre otros).

1.8. Estructura del Informe

La estructura del presente trabajo esta compuesta por 7 capítulos detallados a continuación:

El presente capítulo corresponde a la introducción del trabajo, identificando el problema, la hipótesis de investigación, los objetivos y metodología.

El capítulo 2 representa el marco conceptual para dar al lector un contexto científico en el cual se desarrolla el trabajo.

En el capítulo 3 se muestran las distintas investigaciones que se han realizado en el contexto de la detección de reclamos y la clasificación de contenido en Twitter.

El capítulo 4 describe el procedimiento y los criterios utilizados para la construcción del set de datos que será usado para entrenar los modelos de clasificación de tweets. Se identifican las variables a utilizar y el proceso de recolección de datos.

El capítulo 5 describe la evaluación y la elección de los modelos a utilizar para la clasificación automática de opiniones de reclamos. Se describen las técnicas utilizadas en el pre-procesado del texto. Además se presenta la evaluación de los distintos modelos y algoritmos de clasificación, por medio de la matriz de confusión para cada uno de ellos. Además se explicita el modelo escogido para la identificación de opiniones de reclamos y consultas, y el modelo de categorización de reclamos.

El capítulo 6 presenta el diseño e integración del módulo de reclamos en la pagina web de OpinionZoom. Se describen los elementos y funcionalidades que componen al módulo.

El capítulo 7 presenta las conclusiones generales, en conjunto con recomendaciones y trabajo futuro.

Capítulo 2

Marco Teórico

El presente capítulo pretende dar a conocer algunos elementos importantes con el objetivo de contextualizar al lector y así también familiarizarlo con ciertos conceptos propios del campo de las tecnologías relevantes para una comprensión íntegra del trabajo desarrollado.

2.1. Web 2.0

El término de Web 2.0 nace el año 2004 y se utiliza para caracterizar a los sitios que superan la visión estática en donde los usuarios solo leen el contenido escrito por los creadores de las páginas. En la Web 2.0 los usuarios son una parte activa de los sitios, ahora además son capaces de interactuar y generar su propio contenido [12].

Se calcula que en los años 90 en torno al 80 % del contenido existente en Internet estaba creado por empresas y medios de comunicación y que tan solo el 20 % restante había sido generado por los usuarios (personas quienes visitan las páginas web). En pleno 2006 la situación de la década anterior se había invertido completamente: más del 85 % del contenido existente en la Red estaba creado por usuarios, mientras que las empresas y los medios aparecían relegados a producir en torno al 15 % del total.

La Web 2.0 es la red como una plataforma que permite conectar a todos los dispositivos existentes (computadoras, smartphones, servidores, etc.). Esta plataforma se compone de aplicaciones, las cuales son las que generan la mayoría de las ventajas, la principal radica en que mientras más gente las utilice, mayor cantidad de contenido se genera y más actualizado se mantiene la información de esta plataforma. Además posibilita a que los usuarios consuman y mezclen datos proveniente de múltiples fuentes [13].

La Web 2.0 ha permitido que se genere un entorno en el que lo importante son las personas, se ha establecido un cambio y una influencia cada día mayor en la sociedad y en la economía tal como las conocemos. En la Web de hoy, las personas ya no “leen, hacen clic y callan”, sino que participan activamente, desarrollando una presencia activa, creando una personalidad on-line, comentando, publicando lo que piensan, etc.

Dentro de las características principales de la Web 2.0 se encuentran [12, 13]:

- **La Web como plataforma:** la mayoría de los servicios de la Web 2.0 están basados en que el usuario utiliza la Web como base para su información, lo que posibilita que el usuario pueda acceder a sus datos desde cualquier dispositivo a través de una conexión a internet.
- **Control de los datos:** En esta nueva Web los datos son del usuario y este puede llevárselos cuando desee.
- **Datos generados por el usuario :** El usuario es el elemento activo, es quien produce los datos que pueblan las aplicaciones. El usuario se convierte en el motivo principal en algunos sitios web como lo es el caso de las redes sociales como Twitter.
- **APIs :** Son una pieza diseñada para facilitar la integración de aplicaciones de terceros, proporciona que terceros puedan interactuar de una forma estandarizada. En el caso de Twitter por ejemplo, permite que otras aplicaciones puedan interactuar con sus datos.
- **Interfaz :** El usuario maneja los aspectos visuales de las paginas, da la posibilidad que se pueda cambiar la presentación, lo que generar una libertad de personalización.

En base a las características de la Web 2.0 los servicios generados se pueden clasificar en distintos grupos [13, 14], los cuales son descritos a continuación:

- **Blogs:** Corresponden a sitios web similares a una bitácora, en donde los usuarios generan contenido en forma de *entradas* o *posts* basados en contenido propio que es actualizado a menudo. Son sitios donde los usuarios pueden publicar opiniones, artículos o lo que deseen. Las entradas de los blogs se ordenan cronológicamente, y permiten la función de búsqueda por fecha, tópicos, keywords, etc.
- **RSS (Rich Site Summary o Really Simple Syndication):** Permiten mantener un seguimiento de los blogs favoritos y un consumo mucho mas eficiente de la información. Se basa en archivos XML que reúne la información y los enlaces de los sitios.
- **Wikis:** Son páginas web en las que varios autores pueden colaborar conjuntamente para editar la información y conformar un documento determinado. Además comprende herramientas que facilitan controlar las versiones y regenerar una versión anterior en caso de errores. La más conocida en el mundo es Wikipedia.
- **Foros:** Corresponden a sitios que representan escenarios de diálogos y suelen estar tematizados.
- **Redes Sociales:** Son sitios que permiten estructurar relaciones en los mas diversos ámbitos, desde profesionales hasta personales. Se genera comunicación e intercambio de información entre los usuarios. Ellos son los que generan el contenido del sitio.
- **Comunidades de contenido:** Corresponden a sitios dedicados al almacén y compartición de contenido, como fotos o videos, con usuarios generalmente categorizados en diferentes círculos de intimidad.

2.2. Twitter

Twitter es un sitio Web de *microblogging* creado en el año 2006 [15]. Tiene la principal característica que los usuarios publican mensajes de texto acotados a 140 caracteres de longitud, llamados

tweets. Los usuarios pueden suscribirse a los tweets de otros usuarios, a lo que se llama "seguir" y se les conoce como *friends*; y los usuarios abonados se les conoce como "seguidores" o *followers*. Twitter es una red social que ha tenido un gran crecimiento, posee 313 millones de usuarios activos y soporta mas de 40 idiomas a lo largo del mundo [16].

Como se mencionó, los tweets están compuestos por un máximo de 140 caracteres, por ende existen varios términos y convenciones en los mensajes y en la red de usuarios que los permiten caracterizar [17, 18]:

- **Followers:** Representan a los seguidores de un usuario.
- **Following/Friends:** Corresponden a las cuentas que un usuario está siguiendo.
- **Status/Tweets:** Estos representan las actualizaciones de estado de los usuarios.
- **ReTweet (RT):** Corresponde a publicar nuevamente un tweet, la acción se denomina *retwittear*.
- **Hashtags (simbolo #):** Representan las etiquetas (son palabras escritas con el símbolo # antepuesto) que se usan para indexar palabras claves o temas en Twitter.
- **Usuario (símbolo @):** Se usa para indicar a usuarios dentro de un tweet. Los usuarios que se nombren en un tweet recibirán dicho mensaje.
- **Mensajes Directos / Direct Messages (DM):** Son mensajes privados que se envían entre usuarios de Twitter. Se utilizan para mantener conversaciones privadas con un solo usuario o con grupos de usuarios.
- **Menciones:** Corresponden a tweets en donde el mensaje contiene @nombredeusuario en el cuerpo del mensaje. El usuario mencionado recibe dicho mensaje.
- **Respuestas:** Corresponden a mensajes asociados a otros. Son similares a las menciones, pero se diferencian en que @nombredeusuario se encuentra en el comienzo del tweet.

2.2.1. APIs de Twitter

Twitter posee varias API's[1], pero dos de estas, la Streaming API y la REST API, sirven para operar con los tweets. La primera permite interactuar con los datos que se generan en tiempo real en Twitter, mientras que la segunda permite interactuar con los datos históricos[2]. Además la REST API tiene la capacidad de realizar cambios (publicar tweets, responder tweets, modificar datos del usuario, etc.) para la cuenta que se esté utilizando.

El soporte para la REST API se encuentra de manera extra oficial en diversos lenguajes, siendo uno de estos JAVA con la librería Twitter4J. En la tabla2.1 se muestran algunos [3] de los recursos de la librería Twitter4J.

Los datos que entregan las funciones de la librería de Twitter4J se encuentran en formato JSON, acrónimo de JavaScript Object Notation. JSON es un formato de texto ligero para el intercambio

[1]La interfaz de aplicaciones (abreviada API del inglés), es el conjunto de subrutina, funciones y procedimientos que ofrece cierta biblioteca para ser utilizada por otro software como una capa de abstracción.

[2]presenta ciertas limitaciones. Para algunas funciones de la API tan solo permite rescatar una cantidad establecida de datos.

[3]http://twitter4j.org/en/api-support.html

Recurso de Twitter4J	Descripción
Timelines	Funciones que entregan tweets recientes de un usuario
Tweets	Funciones para un tweet específico
Search	Entrega tweets que coinciden con una consulta específica
Streaming	Funciones para interactuar con la Streaming API
Direct Message	Funciones para operar con mensajes directos
Friends & Followers	Funciones para los seguidores y amigos
Users	Funciones para operar con la información de un usuario
Favorites	Funciones para obtener información de los favoritos
Places & Geo	Funciones para obtener datos sobre localización
Trends	Obtener información acerca de los trending topics
Help	Obtener información de configuración y ayuda

Tabla 2.1: Detalle recursos de Twitter4J.
Fuente: Elaboración propia.

de datos [19], es fácil de entender e interpretar por humanos y máquinas. El formato JSON es completamente independiente del lenguaje programación, pero utiliza convenciones que son familiares a muchos de ellos, entre los que se encuentra JAVA.

El formato JSON está construido en base a dos estructuras:

- Un conjunto de pares llave/valor.
- Una lista ordenada de valores.

Estas dos estructuras de datos son universales, prácticamente todos los lenguajes de programación modernos los soportan de una forma u otra.

2.3. Knowledge Database Discovery - KDD

El proceso KDD o Proceso de Extracción de Conocimiento de Bases de Datos en español, se refiere a la totalidad de etapas que se deben llevar a cabo para descubrir patrones en los datos. Estos pasos en el proceso KDD permiten cerciorarse que los patrones provienen de los datos. La aplicación a ciegas de métodos de minería de datos pueden ser una actividad riesgosa, ya que puede fácilmente conducir al descubrimiento de patrones sin sentido e inválidos.

El proceso KDD consta de 9 etapas como se describe en [20]:

1. **Desarrollar un entendimiento del dominio de aplicación:** Este es el paso preparatorio que establece el escenario para entender lo que se debe hacer con: la transformación, los algoritmos y la representación. Se necesita entender y definir: los objetivos del usuario final, donde se llevará a cabo el proceso, y otros conocimientos previos relevantes.

2. **Selección y creación de un conjunto de datos en función de los objetivos:** Determinar qué datos se utilizarán, tales como: los datos disponibles, datos adicionales necesarios y la inte-

gración de todos en un conjunto único de datos, incluidos los atributos que se considerarán para el proceso. Este punto es crucial debido a que la minería de datos aprende a partir de los datos disponibles. Esta es la base para construir los modelos. Si faltan algunos atributos importantes, entonces todo el estudio puede fallar o generar resultados incoherentes. Desde este punto de vista, cuantos más atributos se consideren, mejor.

3. **Pre-procesamiento y limpieza de datos:** Este paso tiene por finalidad mejorar la fiabilidad de los datos. Incluye la eliminación de datos, como el manejo de valores omitidos o faltantes, y la eliminación de valores atípicos. Puede implicar complejos métodos estadísticos, o utilización de un algoritmo de minería de datos en este contexto.

4. **Transformación de datos:** En esta etapa, se prepara y desarrolla la generación de mejores datos. Los métodos utilizados en este paso incluyen la reducción de dimensionalidad, selección de características, extracción, muestreo de datos, y transformación de atributos como discretización a atributos numéricos. Este paso puede determinar el éxito de todo el proceso de KDD, y por lo general es muy específico del proyecto. Una vez completados los cuatro pasos anteriores, los siguientes cuatro pasos se relacionan con la minería de datos, que se centra en los aspectos algorítmicos empleados para cada proyecto.

5. **Escoger la tarea de Data Mining adecuada:** Se debe decidir qué tipo de estrategia de Data Mining utilizar. Por ejemplo: clasificación, regresión o agrupación. Esta elección depende específicamente de los objetivos del proceso KDD, y de los pasos anteriores. Existen dos propósitos principales en la minería de datos: predicción y descripción. La predicción se refiere a la minería de datos supervisada, mientras que la minería de datos descriptivos incluye los aspectos no supervisados y de visualización de la minería de datos.

   La mayoría de las técnicas de minería de datos se basan en el aprendizaje inductivo, donde un modelo se construye explícita, o implícitamente, generalizando a partir de un número suficiente de ejemplos de entrenamiento. La suposición subyacente del enfoque inductivo es que el modelo entrenado es aplicable a casos futuros. La estrategia también tiene en cuenta el nivel de meta-aprendizaje para el conjunto particular de datos disponibles.

6. **Escoger el algoritmo de Data Mining:** Ya determinada la estrategia, podemos decidir qué tácticas utilizar. Esta etapa incluye seleccionar el método específico para buscar patrones, esta elección depende en específico del problema que se quiera resolver, por ejemplo existen algoritmos de clustering, que permiten agrupar datos con características similares, algoritmos de clasificación, que permiten identificar una clase, entre otros.

7. **Aplicación de algoritmo de Data Mining:** Finalmente, se puede implementar el algoritmo de minería de datos escogido. Es necesario aplicar el algoritmo reiteradas veces hasta que se obtenga un resultado satisfactorio. Por ejemplo, modificar los parámetros de control de algoritmos para analizar variaciones en los resultados y escoger los mejores.

8. **Evaluación:** En esta etapa se evalúan e interpretan los patrones encontrados con respecto a los objetivos definidos en la primera etapa. Se consideran los pasos del pre-procesamiento de datos con respecto a su efecto en los resultados del algoritmo de Data Mining. El último paso es el uso y la retroalimentación general sobre los patrones y resultados obtenidos.

9. **Usando el conocimiento descubierto:** Como paso final se debe incorporar el conocimiento para futuras acciones. El conocimiento se vuelve activo en el sentido de que podemos hacer cambios en el sistema y medir los efectos. Este paso determina la eficacia de todo el proceso KDD. Existen bastantes desafíos en este paso, como la pérdida de condiciones bajo las cuales se operó. Por ejemplo, el conocimiento se descubrió a partir de una cierta situación instantánea estática, pero ahora los datos se vuelven dinámicos. Las estructuras de datos pueden cambiar y el dominio de datos puede modificarse.

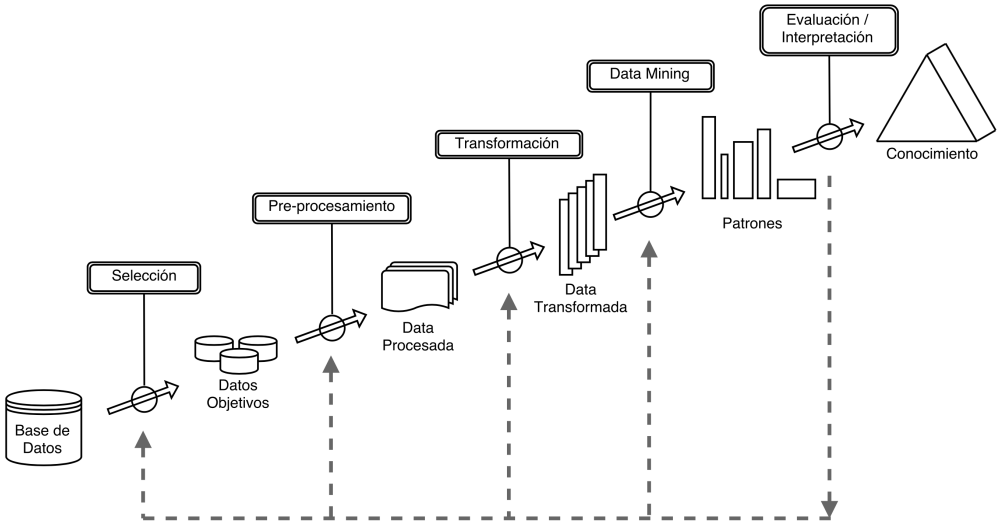

Figura 2.1: Proceso KDD.
Fuente: Elaboración propia.

2.3.1. Data Mining

Tal como se mencionó anteriormente, Data Mining corresponde a un paso dentro del proceso KDD. El termino Data Mining o minería de datos en español, es un área de la estadística y las ciencias de la computación que busca descubrir patrones en grandes volúmenes de datos. Hace uso de métodos de inteligencia artificial, aprendizaje automático, estadísticas y sistema de bases de datos. El propósito general del proceso de minería de datos consiste en extraer información relevante de un conjunto de datos y transfórmalo en una estructura comprensible que permita un análisis y uso posterior.

Es necesario aclarar que si bien las técnicas de Data Mining permiten identificar patrones en los datos, existen varias otras etapas que contemplan: la recolección de datos, preparación de estos mismos e interpretación de resultados; que son previas a Data Mining y pertenecen al proceso KDD [21].

2.4. Text Mining

Text Mining, también conocido como minería de texto o descubrimiento de conocimiento a partir de bases de datos textuales, se refiere generalmente al proceso de extraer patrones interesantes y no triviales o conocimiento de documentos de texto no estructurados. Puede ser visto como una extensión de Data Mining o KDD, ya que utiliza una estructura similar para resolver problemas, con la salvedad de que utiliza técnicas específicas para tratar con texto. La minería de texto es un campo que intenta recopilar información significativa a partir del texto del lenguaje natural. Puede ser vagamente caracterizado como el proceso de analizar el texto para extraer información que es útil para propósitos particulares [22]. En comparación con el tipo de datos almacenados en las bases de datos, el texto es no estructurado, amorfo, y difícil de tratar con algoritmos sin antes procesarlo a una estructura entendible para los modelos y algoritmos.

A primera vista, Data Mining parece bastante similar a Text Mining, sin embargo poseen grandes diferencias. La minería de datos se puede caracterizar más completamente como la extracción de información implícita, previamente desconocida y potencialmente útil de los datos. La información está implícita en los datos de entrada: está oculta, desconocida y difícilmente podría extraerse sin recurrir a técnicas automáticas de minería de datos. Por otro lado, con la minería de textos, la información que se va a extraer es clara y está explícitamente en el texto. No se oculta en absoluto, y desde un punto de vista humano, el único sentido en el que es "desconocido" es que las restricciones de recursos humanos hacen que sea imposible que las personas lean y analicen todo el texto. El problema recae entonces, en que la información no está redactada de una manera que sea susceptible para el procesamiento automático de una máquina, para esto, Text Mining provee una serie de herramientas que permiten procesar el texto a una forma que sea adecuada para el consumo de las computadoras [23], sin necesidad de un intermediario humano.

2.4.1. Procesamiento de texto

La minería de textos tiene un valor comercial muy alto. Es una tecnología emergente para analizar grandes colecciones de documentos no estructurados con el fin de extraer un patrón o conocimiento interesante y no trivial [24]. Existen muchas aplicaciones específicas del dominio de Text Mining, algunas de estas son:

- Análisis del perfil del cliente
- Aplicaciones de seguridad
- Aplicación biomédica
- Planificación de la empresa
- Respuestas a encuestas abiertas
- Inteligencia competitiva
- Gestión de relaciones con clientes (CRM)
- Organizar los repositorios de documentos relacionados con meta-información

Los seres humanos tienen la capacidad de distinguir y aplicar patrones lingüísticos al texto y pueden superar con facilidad los obstáculos que las computadoras no pueden manejar fácilmente,

como la jerga, las variaciones ortográficas y el significado contextual. Sin embargo, aunque las capacidades de lenguaje permiten entender los datos no estructurados, a las personas les falta la capacidad de procesar el texto en grandes volúmenes y a altas velocidades. Por lo tanto, la minería de texto ayuda a las computadoras en la tarea de análisis de datos no estructurados. Para esto existen una serie de técnicas para tratar con texto [24]. A modo de ejemplo se utilizará el siguiente set de dos documentos para la explicación de las cuatro técnica de preprocesamiento de texto.

(1) *@AyudaMovistarCL me encuentro sin servicio de internet*
(2) *siempre falla el servicio de televisión @AyudaMovistarCL*

2.4.1.1. Tokenización

El primer paso para preprocesar el texto tiene el objetivo de subdidivr los documentos a elementos individuales que puedan caracterizar al texto, proceso llamado *Tokenización*. Típicamente, la tokenización ocurre en el nivel de la palabra. Sin embargo, a veces es difícil definir lo que se entiende por una "palabra". A menudo un tokenizador se basa en algoritmos heurísticos simples, por ejemplo:

- La puntuación y espacios en blanco pueden incluirse o no en la lista resultante de fichas.
- Todas las cadenas contiguas de caracteres alfabéticos forman parte de un token, así también los números y la combinación de ambos.
- Los símbolos que están separados por caracteres de espacio en blanco, como un espacio o un salto de línea, o por caracteres de puntuación.

Basado en los dos documentos de ejemplo, la lista de tokens del set se construye como sigue:

```
[
    “@AyudaMovistarCL”,
    “me”,
    “encuentro”,
    “sin”,
    “servicio”,
    “de”,
    “internet”,
    “siempre”,
    “falla”,
    “el”,
    “televisión”
]
```

2.4.1.2. Borrado de Stopwords

Este proceso se encarga de remover palabras que no aportan un valor significativo en el texto. Generalmente representan a las palabras más comunes del idioma, que poseen una alta frecuencia

en los documentos, pero que no agregan información relevante al contexto más allá de su función gramatical. Dentro de estas palabras se encuentran: artículos, preposiciones, conjunciones, pronombres, entre otras. Además no existe ninguna lista universal única de *StopWords*, sin embargo un buen ejemplo de esta se encuentra en [25] para el idioma español, la cual cuenta con 325 StopWords.

Para el ejemplo propuesto los tokens eliminados en esta etapa serían: “me”, “de” y “el”.

2.4.1.3. Stemming

El proceso de *Stemming* consiste en reducir las palabras a su forma base o raíz, cuya forma base no necesariamente es idéntica a la raíz morfológica de la palabra. Este algoritmo se lleva a cabo eliminando los afijos de las palabras (elementos adicionales adjuntos para cambiar su sentido gramatical). Esto convierte a todas las palabras que tengan una misma raíz a una forma base común.

Tomando el token *falla* del ejemplo propuesto, se puede realizar el algoritmo de *Stemming* dando como resultado el token *fall*. Además como se muestra a continuación todas las conjugaciones del verbo fallar terminan con la misma raíz tras aplicar el algoritmo de *Stemming*:

$$\left\{ \begin{array}{c} falla \\ fallar \\ fallando \\ falló \\ fallas \end{array} \right\} fall$$

2.4.1.4. N-grams

En el procesamiento de lenguaje natural, un n-gram es una secuencia contigua de *n* tokens dentro de un texto. Estos tokens pueden ser fonemas, sílabas, letras, palabras o pares de estos de acuerdo a la aplicación. Cuando estos elementos son palabras, los n-grams se denominan *shingles* [26].

Un n-gram de tamaño 1 se denomina *unigram*; El tamaño 2 es un *bi-gram* (o digram); El tamaño 3 es un *tri-gram*, y así sucesivamente.

La utilización de n-grams es utilizada en el procesamiento de lenguaje natural para incorporar contexto a la variables. Se trata de abarcar un conjunto de token consecutivos de modo de captar una porción mayor del texto. Sin embargo, no existe una regla que diga la cantidad óptima de n-grams a utilizar, el tamaño siempre depende de los datos que se tengan, por esto, no siempre una mayor cantidad de n-grams mejora el rendimiento de los algoritmos de Machine Learning en la clasificación de texto [27].

Para el ejemplo propuesto los bi-grams resultantes serian (habiendo removido las Stopwords, pero sin realizar Stemming):

[
“@AyudaMovistarCL_encuentro”,
“encuentro_sin”,
“sin_servicio”,
“servicio_internet”,
“siempre_falla”,
“falla_servicio”,
“servicio_televisión”,
“televisión_@AyudaMovistarCL”,
]

2.4.2. Bag-Of-Words

Dentro de las estructuras más conocidas y utilizadas para representar un set de documentos de textos se encuentra *Bag-Of-Words* [28], que se encarga de modelar un set de documentos para que pueda ser comprendido por un computador. La idea clave de este modelo es medir la ocurrencia de cada palabra de un set de documentos para una instancia particular. En muchos casos la aplicación de la ocurrencia de términos no es la mejor representación del texto, por lo que existen otras formas de abordar este problema, como lo es la frecuencia de terminos y TF-IDF (Term Frecuency - Inverse Document Frecuency).

TF-IDF es una estadística numérica que pretende reflejar la importancia de una palabra para un documento en una colección [29]. El valor TF-IDF aumenta proporcionalmente al número de veces que una palabra aparece en el documento, pero se compensa con la frecuencia de la palabra en el set de documentos, lo que ayuda a ajustar el hecho de que algunas palabras aparecen más frecuentemente en general. Este valor se calcula como lo muestra la ecuación 2.1

$$TFIDF(t,\mathrm{d}) = TF(t,\mathrm{d}) \times IDF(t) \tag{2.1}$$

donde,

$$TF(\mathrm{d},t) = \textit{Frecuencia de término t en documento d} \tag{2.2}$$

$$IDF(t) = 1 + log\left(\frac{\textit{Número total de documentos}}{\textit{Documentos con término t}}\right) \tag{2.3}$$

Tomando estos tres modos de ponderar las palabras dentro de un set de datos, se puede construir el vector para cada instancia como muestra a continuación, considerando el ejemplo provisto en páginas anteriores.

Esta forma de representación de texto es la adecuada como *input* para los modelos clasificadores que serán descritos en la siguiente sección.

(1) [1 , 1 , 1 , 1 , 1 , 1 , 0 , 0 , 0 , 0]
(2) [1 , 0 , 0 , 0 , 1 , 1 , 1 , 1 , 1 , 1]

2.5. Machine Learning

Machine Learning, Aprendizaje automático o Aprendizaje de Maquinas, es un tipo de inteligencia artificial (IA) que proporciona a los computadores la capacidad de aprender sin ser programados de forma explicita. Machine Learning se define como el proceso de programar un máquina o computador para que sea capaz de entregar resultados lo suficientemente útiles en base al uso de datos de ejemplo o experiencias pasadas [30]. Además se centra en el desarrollo de programas informáticos que pueden enseñarse a crecer y cambiar cuando se exponen a nuevos datos.

El proceso de aprendizaje automático es similar al de la minería de datos. Ambos sistemas buscan patrones a través de datos, sin embargo, en lugar de extraer patrones para la comprensión humana, como es el caso de las aplicaciones de minería de datos, el aprendizaje automático utiliza esos datos para detectar patrones y ajustar las acciones del algoritmo en cuestión.

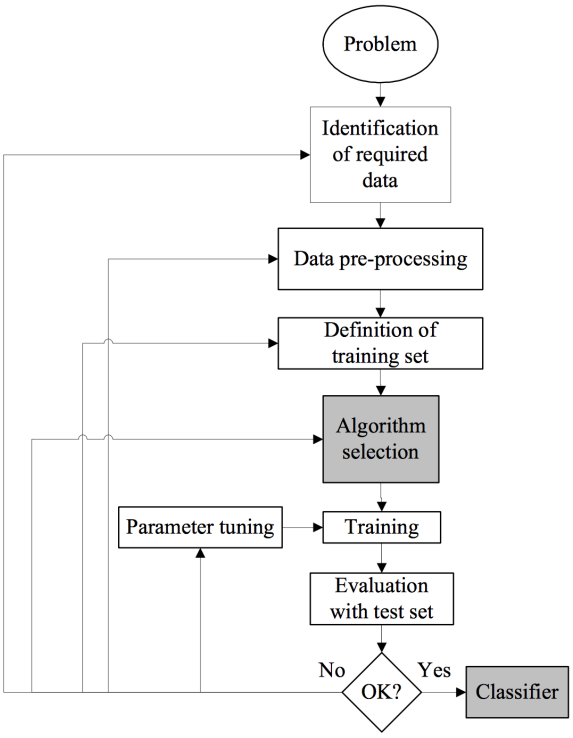

Figura 2.2: Aspectos generales de los algoritmos de aprendizaje supervisado.
Fuente: Supervised Machine Learning [11].

Los algoritmos de aprendizaje automático suelen clasificarse en tres grandes categorías: aprendizaje supervisado, aprendizaje no supervisado y aprendizaje reforzado. Los algoritmos supervisados pueden aplicar lo que se ha aprendido en el pasado con los nuevos datos, los algoritmos no supervisados pueden extraer inferencia de los conjuntos de datos y los algoritmos reforzados descubren vía prueba y error las acciones que conllevan a mejores resultados.

La figura 2.2 muestra la metodología que se debe llevar a cabo para la aplicación de un modelo de Machine Learning supervisado, el cual consta de 9 etapas y se detalla a continuación:

1. **Problema:** En una primera instancia se debe identificar el problema que se pretende resolver, para el caso de esta memoria, este corresponde a la identificación y categorización de opiniones de reclamos en textos provenientes de Twitter.

2. **Identificar datos necesarios:** Como segundo paso se debe definir que datos son los que se utilizarán, y de donde y como se extraerán. Siguiendo la analogía con lo desarrollado, esto corresponde a identificar específicamente que *tweets* serán considerados para formar parte del set de datos y que métodos de la API de Twitter se utilizarán.

3. **Selección y preprocesamiento de datos:** De todos los datos necesarios recolectados, se deben seleccionar solo aquellos que se utilizarán para entrenar el modelo. Aquí también aplican herramientas de Data Mining para seleccionar los datos más relevantes, por ejemplo remover datos nulos, datos redundantes, entre otros. Además se debe realizar, de ser necesario, el proceso de etiquetado de los datos, ya que se utilizarán algoritmos supervisados. Finalmente, se hace uso de técnicas de preprocesamiento de datos como: *tokenización*, *StopWords*, *Stemming*, *POS-Tagger*, *lematización*, entre otros.

4. **Definición del set de entrenamiento:** Como cuarta etapa se debe definir el set de entrenamiento final para el algoritmo de Machine Learning, se debe realizar una división del set de datos en dos: uno para entrenar el modelo y otro la testearlo y obtener las métricas de desempeño

5. **Selección de algoritmo:** Se debe escoger algún algoritmo de Machine Learning supervisado. Además para el desarrollo de esta memoria, este algoritmo debe tener la capacidad de clasificar en múltiples clases, lo que se denomina clasificador *multi-clase*.

6. **Entrenamiento:** Habiendo seleccionado el algoritmo, se debe entrenar el modelo, para esto se debe particionar el set de entrenamiento en dos conjuntos: uno para entrenar el modelo y otro para validarlo.

7. **Evaluación con set de validación:** Una vez entrenado el modelo, se debe evaluar que tan eficaz es. Para ello se utiliza el set de validación o prueba, y los resultados se muestran en una matriz de confusión para el calculo de los indicadores de *accuracy, precision, recall y f-measure*.

8. **Ajuste de parámetros:** Si los resultado del algoritmo diseñado no son buenos se puede volver a los pasos anteriores con el objetivo de variar alguno de estos y analizar posibles mejoras al algoritmo.

9. **Clasificador:** Finalmente si los resultados son acorde a los esperado se concluye con el clasificador.

Dentro de esta metodología descrita es posible identificar dos grandes sub-procesos: (1) Diseño y Construcción del set de datos / Set de entrenamiento y (2) Evaluación de Algoritmos de Machine Learning. El primero contempla a los pasos 2, 3 y 4, y el segundo, los pasos 5, 6, 7 y 8.

Además de la categorización antes mencionada sobre los algoritmos de aprendizaje de máquina, existe otra clasificación en función del *output* deseado [31], para lo cual se logran identificar los siguientes enfoques:

- **Clasificación:** Los insumos o datos de entrenamiento se dividen en dos o más clases, y el algoritmo debe producir un modelo que asigna entradas no vistas a una o más (clasificación de etiqueta múltiple) de estas clases. Esto se aborda típicamente de una manera supervisada. El filtrado de correo no deseado es un ejemplo de clasificación, donde las entradas son mensajes de correo electrónico (u otros) y las clases corresponden a"spam" y"no spam".
- **Regresión:** Es un modelo que también utiliza algoritmos supervisados, pero en este caso las salidas o *outputs* corresponden a valores continuos en lugar de discretos.
- **Agrupación:** En este caso el conjunto de datos de entrada o *inputs* se divide en grupos. A diferencia de la Clasificación primeramente descrita, los grupos no se conocen de antemano, por lo que esta tarea es de carácter no supervisada.
- **Estimación de Densidad:** Estos modelos permiten determinar la distribución de los *inputs* dentro de algún espacio. Corresponde a la construcción de una estimación basada en los datos observados. La función de densidad no observable se considera como la densidad según la cual se distribuye una gran población; Los datos suelen ser considerados como una muestra aleatoria de esa población.
- **Reducción de Dimensionalidad:** Estos modelos simplifican los *inputs* asignándolos a un espacio de menor dimensión. Topic Modeling o modelo de tópicos, es un problema relacionado, en el que se le entrega a un algoritmo una lista de documentos en lenguaje humano (documento en forma de texto) y se le encarga averiguar qué documentos cubren temas similares. Cabe destacar que estos son modelos no supervisados.

2.5.1. Algoritmos Supervisados

Los algoritmos de aprendizaje supervisados son entrenados usando datos etiquetados, tales como una entrada o *input* en donde se conoce la salida o *output* deseado. Por ejemplo, se podría tener un conjunto de datos rotulados como "F" (fallado) o "E" (ejecutado). El algoritmo de aprendizaje recibe este conjunto de entradas en conjunto con las salidas correspondientes, y el algoritmo aprende comparando su *output* (el que entrega el algoritmo) con los *outputs* verdaderos (las etiquetas o rótulos de los datos) para encontrar errores. A continuación, modifica los parámetros del modelo en consecuencia para encontrar el modelo que permita predecir de mejor forma los datos. Mediante diferentes métodos supervisados como clasificación, regresión, predicción y aumento de gradiente; se utilizan patrones para predecir los valores de la etiqueta en los datos no vistos. El aprendizaje supervisado es comúnmente usado en aplicaciones donde los datos históricos predicen eventos futuros. Por ejemplo, puede anticipar cuando es probable que las transacciones con tarjetas de crédito sean fraudulentas o que el cliente de seguros pueda presentar un reclamo.

En términos de algoritmos supervisados para clasificación de texto en más de dos tópicos, existen distintos modelos [11], dentro de los más utilizados en *Topic Classification* aplicado en Twitter se encuentran: Multinomial Naive Bayes, Multi-Class Support Vector Machines, Artificial Neural Networks, Multinomial Logistic Regression y Decision Trees. A continuación se describen estos algoritmos:

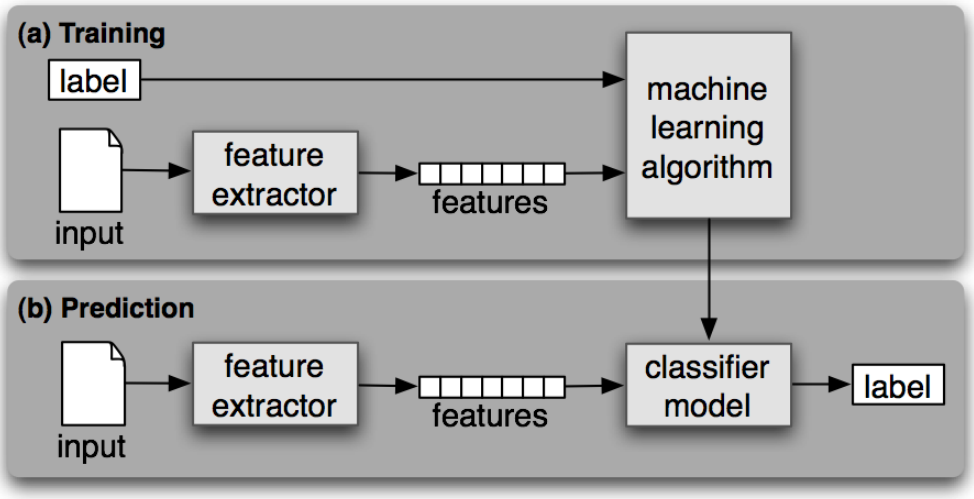

Figura 2.3: Esquema de un Algoritmo Supervisado.
Fuente: S. Bird *et al*. Natural Language Processing with Python [32].

2.5.2. Multinomial Naive Bayes

Multinomial Naive Bayes (MNB) es una de las dos variantes clásicas de Naive Bayes que está diseñada para predecir más de dos clases en la clasificación de documentos de texto (donde los datos son típicamente representados como un recuento vectorial de palabras, aunque también se sabe que los vectores TF-IDF funcionan bien en la práctica). Mientras que Naive Bayes modelaría un documento como la presencia y la ausencia de palabras particulares, MNB modela explícitamente los conteos de las palabras.

Multinomial Naive Bayes es un modelo supervisado que permite calcular la probabilidad de pertenencia a una clase para un documento dado [33]. La ecuación 2.4 muestra el cálculo de pertenencia a una clase c para un término t_i. El valor C determina el conjunto total de clases para los documentos y N el tamaño total de palabras del set de documentos. MNB asigna el termino t_i a la clase que posea la mayor probabilidad descrita por la formula:

$$\mathbb{P}(c|t_i) = \frac{\mathbb{P}(c)\mathbb{P}(t_i|c)}{\mathbb{P}(t_i)}, c\varepsilon C \tag{2.4}$$

2.5.3. Multi-Class Support Vector Machines (SVM)

Support Vector Machines (SVM) es la técnica de aprendizaje supervisado automático más reciente [34]. SVM fue diseñado originalmente para separar clases binarias (k = 2) con un criterio de margen máximo, sin embargo, los problemas del mundo real, así también el abordado en esta memoria, requieren la discriminación en más de dos categorías. En la práctica, los problemas de clasificación de varias clases (k >2) se descomponen comúnmente en una serie se problemas binarios, tales que el modelo de SVM clásico puede aplicarse directamente. Existen dos principales enfoques para enfrentarse a un problema de multi-clases por medio de SVM, estos son one-versus-rest

(1VR) [35] y one-versus-one (1V1) [36] que serán explicados más adelante. Ambos descomponen el problemas de varias clases en un conjunto predefinido de problemas binarios. Este tipo de modelos combina múltiples problemas de optimización de clase binaria en una sola función objetivo y alcanza simultáneamente la clasificación de varias clases. Sin embargo, se requiere una mayor complejidad computacional para el tamaño del problema de programación cuadrática resultante.

El problema que resuelve el algoritmo clásico de SVM es encontrar el hiper-plano óptimo para separar dos clases binarias siguiendo el criterio de margen máximo. Dado los vectores de entrenamiento $x_{\mathrm{i}} \in R^{\mathrm{d}}, \mathrm{i} = 1, ..., l$, en dos clases, y el vector de etiquetas $y \in 1, -1^{l}$, El algoritmo de SVM resuelve el siguiente problema de optimización:

$$\begin{aligned} \underset{w \in H, b \in R, \xi_{\mathrm{i}} \in R}{\text{minimize}} \quad & \frac{1}{2} w^{T} w + C \sum_{\mathrm{i}=1}^{l} \xi_{\mathrm{i}} \\ \text{s.a} \quad & y_{\mathrm{i}} (w^{T} \varphi(x_{\mathrm{i}}) + b) \geq 1 - \xi_{\mathrm{i}} \\ & \xi_{\mathrm{i}} \geq 0, \mathrm{i} = 1, ..., l. \end{aligned} \tag{2.5}$$

Donde $w \in R^{\mathrm{d}}$ es el vector de pesos, $C \in R^{+}$ es la constante de regularización y la función de mapeo φ proyecta los datos de entrenamiento en un espacio de características adecuado H para permitir superficies de decisión no lineales.

Como se comentó anteriormente, para aplicar SVM a un problema de múltiples clases existen dos enfoques, los que se describen a continuación.

2.5.3.1. Enfoque One-Versus-Rest

El enfoque de one-versus-rest (1VR) [35] construye *k* clasificadores binarios separados para la clasificación de k-clases. El m-ésimo clasificador binario es entrenado usando los datos de la clase m-ésima como ejemplos positivos y las clases restantes k-1 como ejemplos negativos. Durante la prueba, la etiqueta de clase se determina por el clasificador binario que da el valor de salida máximo. Un problema importante del enfoque de uno contra el resto es el conjunto de entrenamiento desequilibrado. Supongamos que todas las clases tienen un tamaño igual de ejemplos de entrenamiento, la proporción de ejemplos positivos a negativos en cada clasificador individual es *1 / (k-1)*. En este caso, se pierde la simetría del problema original.

2.5.3.2. Enfoque One-Versus-One

Otro enfoque para la clasificación multi-clase es el enfoque one-versus-one (1V1) o la descomposición *pairwise* [36]. Este método evalúa todas las posibles combinaciones de clasificadores por pareja y, por tanto, induce *k (k - 1) / 2* clasificadores binarios individuales. La aplicación de cada clasificador a un dato de prueba genera un voto a la clase ganadora. Un ejemplo de prueba se etiqueta a la clase con más votos. El tamaño de los clasificadores creados por el enfoque de one-versus-one es mucho mayor que el del enfoque de one-versus-rest, sin embargo, el tamaño del problema cuadrático en cada clasificador es menor, lo que hace posible entrenar rápidamente

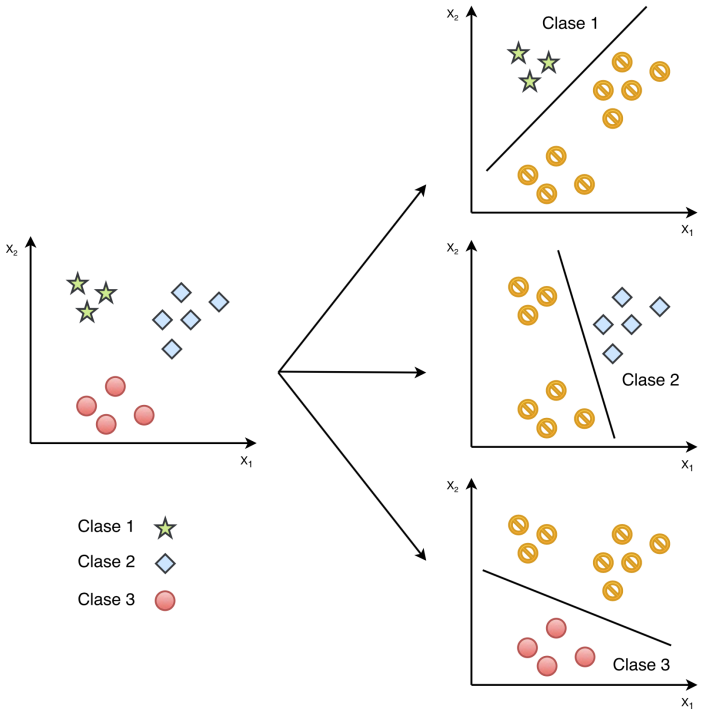

Figura 2.4: Support Vector Machines - Enfoque One-Versus-Rest
Fuente: Elaboración propia.

mediante este enfoque. Además, en comparación con el enfoque de one-versus-rest, este método es más simétrico.

2.5.4. Artificial Neural Networks

Una red neuronal artificial (ANN) en español, o también conocido como Perceptrones multicapa, consiste en un gran número de unidades (neuronas) vinculadas entre sí siguiendo un patrón de conexiones como se aprecia en la figura 2.6. Las neuronas en una red suelen estar segregadas en tres clases: unidades de entrada (input layer), que reciben la información a procesar; Unidades de salida (output layer), donde se encuentran los resultados del procesamiento; Y unidades intermedias conocidas como unidades ocultas (hidden layers).

La red se entrena en un conjunto de datos emparejados para determinar la asignación de entradas y salidas. Los pesos de las conexiones entre las neuronas se fijan y la red se utiliza para determinar las clasificaciones de un nuevo conjunto de datos.

Durante la clasificación, la señal en las neuronas de entrada se propaga completamente a través de la red para determinar los umbrales de activación en todas las neuronas de salida. Cada neurona de entrada tiene un umbral que representa alguna característica externa a la red. Entonces, cada neurona de entrada envía su umbral de activación a cada una de las neuronas ocultas a las que está conectada. Cada una de estas neuronas ocultas calcula su propio umbral de activación y esta señal se pasa a neuronas de salida. El umbral de activación para cada neurona se calcula de acuerdo

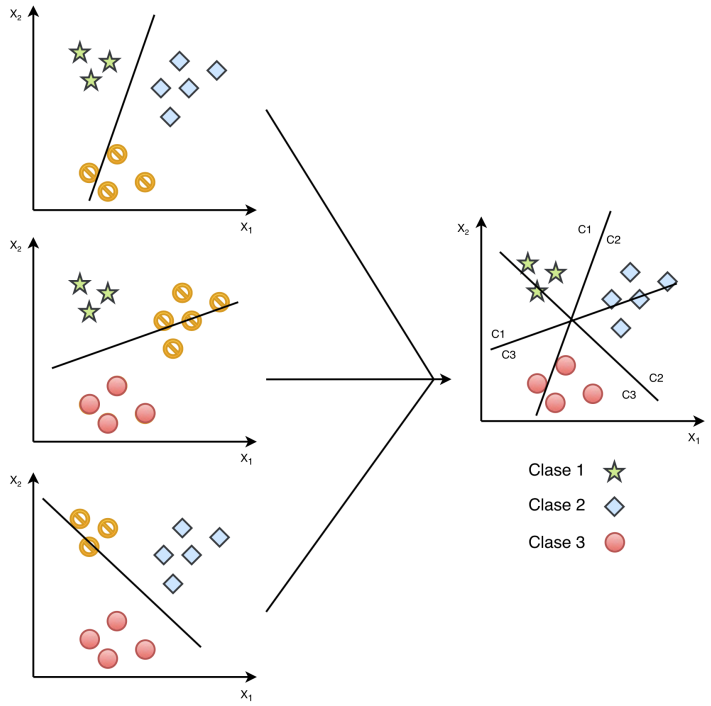

Figura 2.5: Support Vector Machines - Enfoque One-Versus-One
Fuente: Elaboración propia.

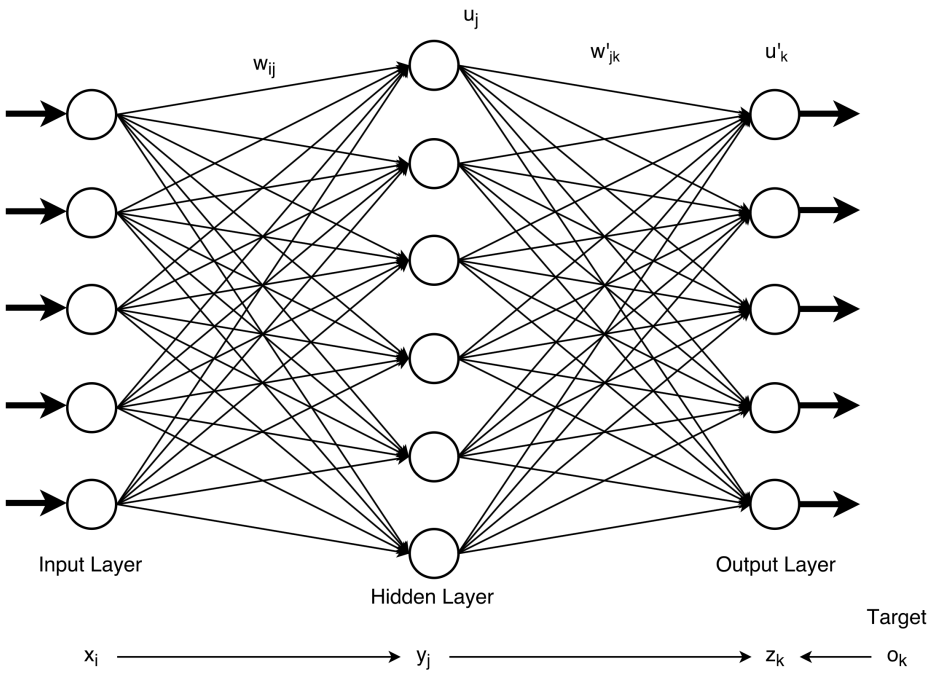

Figura 2.6: Ejemplo de Red Neuroral Artificial con una capa oculta.
Fuente: Elaboración propia.

con una función. Esta función suma las contribuciones de todas las unidades emisoras, donde la contribución de una neurona se define como el peso de la conexión entre las neurona emisora y

receptora multiplicada por el umbral de activación de la neurona remitente.

Generalmente, determinar correctamente el tamaño de la capa oculta es un problema, ya que una subestimación del número de neuronas puede conducir a una pobre aproximación y capacidades de generalización, mientras que excesivos nodos pueden resultar en una sobrecarga y dificultar la búsqueda del óptimo global.

ANN depende de tres aspectos fundamentales: las funciones de entrada y el umbral de activación de cada neurona, la arquitectura de red, y el peso de cada conexión de entrada. Dado que los dos primeros aspectos son fijos, el comportamiento de la ANN se define por los valores de los pesos de las neuronas. Los pesos de la red que se va a entrenar se establecen inicialmente en valores aleatorios, y a continuación, las instancias del conjunto de entrenamiento se exponen repetidamente a la red. Los valores para la entrada de una instancia se colocan en las neuronas de entrada y la salida de la red se compara con la salida deseada para esta instancia. Entonces, todos los pesos en la red se ajustan ligeramente en la dirección que acercaría los valores de salida de la red a los valores para la salida deseada. Existen varios algoritmos con los que se puede capacitar una red, sin embargo, el algoritmo de aprendizaje más conocido y ampliamente utilizado para estimar los valores de los pesos, es el algoritmo de Back Propagation (BP) [37]. Generalmente, el algoritmo de BP incluye los siguientes seis pasos:

1. Evaluar una muestra de datos de entrenamiento a la red neuronal.

2. Comparar el *output* de la red con el *output* real. En cada iteración, calcular el error en cada *output* de las neuronas.

3. Para cada neurona, calcular cuál debió haber sido el *output*, y proporcionalmente, cuánto más bajo o más alto el *output* debe ajustarse para que coincida con la salida real de la data de entrenamiento. Este corresponde al error local.

4. Ajustar los pesos de cada neurona para disminuir el error local.

5. Asignar la responsabilidad del error local a las neuronas en la capa anterior, dando mayor responsabilidad a las neuronas conectadas por pesos más grandes.

6. Iterar los pasos anteriores sobre las neuronas en la capa anterior.

En mayor detalle, la regla general para modificar los pesos de las neuronas en la red es:

$$\Delta W_{ji} = \eta \delta_j O_i \tag{2.6}$$

Donde,

- η es un número positivo (llamado tasa de aprendizaje), que determina el tamaño del paso en la búsqueda de la pendiente del gradiente. Un valor grande permite que la propagación posterior se mueva más rápido a la configuración de peso objetivo, pero también aumenta la posibilidad de que nunca llegue a este objetivo.
- O_i corresponde al *output* calculado por la neurona i.

- $\delta_j = O_j(1-O_j)(T_j-O_j)$ es el valor para el output de la neurona, donde T_j es el valor buscado para la neurona j.
- $\delta_j = O_{\mathrm{i}}(1-O_{\mathrm{i}})\sum_k \delta_k W_{kj}$ es el valor para las neuronas intermedias (hidden layer).

El algoritmo de Back Propagation tiene que realizar una serie de modificaciones en el peso de las neuronas antes de que alcance buenos resultados. Para n instancias de entrenamiento y pesos W, cada iteración en el proceso de aprendizaje toma tiempo $O(nW)$, pero en el peor de los casos, el número de iteraciones puede ser exponencial al número de *inputs*. Por esta razón, las redes neuronales utilizan una serie de reglas de detención para controlar cuando finalizar el algoritmo de entrenamiento. Las cuatro reglas de detención más comunes son:

- Detener después de un número determinado de iteraciones.
- Detener cuando el error entre lo calculado y lo esperado alcanza un umbral predefinido.
- Detener cuando el error no ha visto mejoras en un cierto número de iteraciones.
- Cuando la medida del error en algunas de las instancias que se han analizado a partir de los datos de entrenamiento (training data vs test data) es mayor que una cantidad determinada (overfitting).

2.5.5. Multinomial Logistic Regression

La regresión logística multinomial es conocida por una variedad de otros nombres, incluyendo maximum entropy (MaxEnt) classifier, polytomous LR, multiclass LR, softmax regression, multinomial logit o conditional maximum entropy model.

En estadística, la regresión logística multinomial (MLR) es un método de clasificación que generaliza la regresión logística a problemas de multiclases, es decir, con más de dos posibles resultados discretos. Es decir, es un modelo que se utiliza para predecir las probabilidades de los diferentes resultados posibles de una variable dependiente categóricamente distribuida dado un conjunto de variables independientes (que pueden ser de valor real, binarias, categóricas, etc.). Esta basado en el principio de máxima entropía desarrollado por Maxwell, Boltzmann y Gibbs [38], y no asume supuestos de independencia condicional de las variables como el clasificador de Naive Bayes.

Existen múltiples formas equivalentes de describir el modelo matemático subyacente a la regresión logística multinomial [39], sin embargo, la idea detrás de todas ellas, como en muchas otras técnicas de clasificación estadística, es construir una función predictora lineal que construye una puntuación a partir de un conjunto de pesos que se combinan linealmente con las variables explicativas (características) de una observación dada usando un producto punto.

$$\mathrm{score}(\mathbf{X}_{\mathrm{i}}, k) = \beta_k \cdot \mathbf{X}_{\mathrm{i}} \tag{2.7}$$

Donde $\mathbf{X}_{\mathrm{i}}$ es el vector de las variables explicativas de la instancia i, β_k es un vector de pesos (o coeficientes de regresión) correspondiente al resultado k, y el valor $\mathrm{score}(\mathbf{X}_{\mathrm{i}}, k)$ es la puntuación asociada con la asignación de la instancia i a la clase k. Finalmente la instancia es asignada a la

clase con la puntuación más alta.

2.5.6. Decision Trees

Los árboles de decisión son algoritmos que clasifican las instancias en función de los valores de sus características. En un árbol de decisión cada nodo representa una característica en una instancia a clasificar y cada rama representa un valor que esta variable puede tomar. Las instancias se clasifican a partir del nodo raíz y se ordenan en función de los valores de sus característica.

Los modelos de árbol donde la variable objetivo puede tomar un conjunto finito de valores se llaman árboles de clasificación; En estas estructuras de árbol, las hojas representan las etiquetas de clase y las ramas representan las diferentes combinaciones o valores que pueden tomar las características y que conducen a esas etiquetas de clase. Los árboles de decisión donde la variable objetivo puede tomar valores continuos (típicamente números reales) se llaman árboles de regresión.

El aprendizaje basado en arboles de decisión es un método comúnmente utilizado en la minería de datos [40], el objetivo es crear un modelo que prediga la clase de una instancia como función de diversas características de la variable de entrada o input. Un ejemplo se puede apreciar en la figura 2.7.

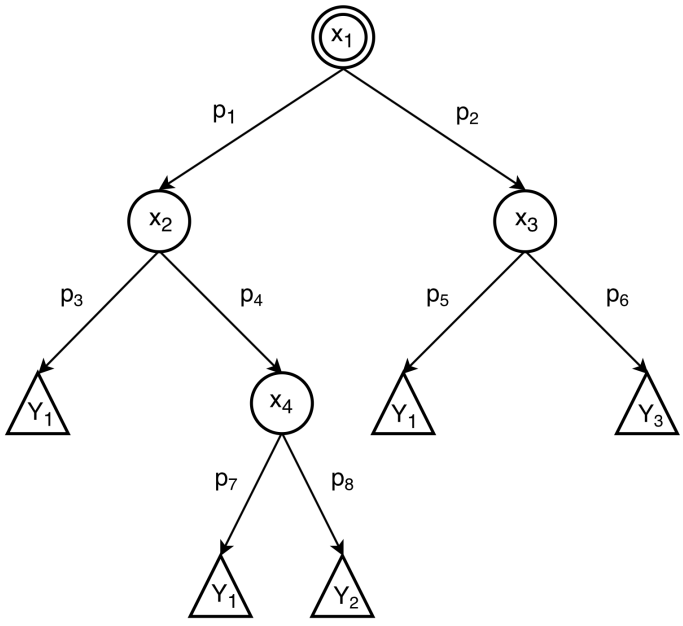

Figura 2.7: Ejemplo de Árbol de Desición.
Fuente: Elaboración propia.

En un árbol de decisión cada nodo interior (x_2, x_3 y x_4, representados por círculos en el ejemplo) corresponde a una de las variables (o atributos) del *input*, las ramas llevan a otros nodos u hojas para cada uno de los posibles valores de esa variable de entrada. Finalmente cada hoja (representada como triángulo en el ejemplo) corresponde a la clase de dicha instancia representada por el camino desde la raíz (nodo x_1) hasta la hoja respectiva.

Un árbol de decisión o un árbol de clasificación es un árbol en el que cada nodo interno (no-hoja) está etiquetado con una característica de la variable de entrada. Los arcos que provienen de un nodo etiquetado con una entidad están marcados con cada uno de los valores posibles de esta variable, y cada hoja del árbol se etiqueta con una clase o una distribución de probabilidad sobre las clases.

Un árbol puede ser entrenado dividiendo el conjunto de instancias en subconjuntos basándose en un valor de algún atributo. Este proceso se repite en cada subconjunto derivado de una manera recursiva, llamada partición recursiva. La recursión se completa cuando el subconjunto en un nodo tiene el mismo valor de la clase buscada o cuando la división ya no agrega valor a las predicciones. Este proceso de inducción descendente de árboles de decisión es un ejemplo de un algoritmo codicioso (greedy algorithm), y es por mucho la estrategia más común para el aprendizaje de árboles de decisión a partir de datos.

La característica o atributo que mejor divide los datos de entrenamiento debe ser el nodo raíz del árbol. Existen numerosos métodos para encontrar la característica que mejor divide los datos de entrenamiento, siendo la más utilizada la *ganancia de información* (conocida en inglés como Information Gain). El criterio de Information Gain está basada en el concepto de entropía y se define como sigue:

$$I_E = -\sum_{\mathrm{i}=1}^{J} f_{\mathrm{i}} log_2 f_{\mathrm{i}} \tag{2.8}$$

Information Gain = Entropía(padre) - Suma Ponderada de Entropía(hijos)

$$IG(T,a) = H(T) + H(T|a) \tag{2.9}$$

Finalmente este procedimiento se repite en cada partición de los datos divididos, creando subárboles hasta que los datos de entrenamiento se dividen en subconjuntos de la misma clase.

2.6. Evaluación de Resultados para modelos de clasificación

Para todo modelo o algoritmo desarrollado es importante medir el desempeño de este, con el objetivo que identificar y comparar sus resultados. En especial para el trabajo de título desarrollado se utilizan métricas de desempeño para algoritmos de clasificación, que se basan en la *Matriz de Confusión*, y métricas de acuerdo entre observadores, basadas en el coeficiente de *Kappa de Fleiss* [41].

2.6.1. Métricas de Desempeño

En Machine Learning, estadística y Data Mining existen una serie de términos comunes que permiten evaluar los modelos. Para un algoritmo de clasificación los resultados se muestran en

una matriz de confusión [42]. La figura 2.2 muestra un ejemplo de resultados de un algoritmo de clasificación de dos clases.

		Clases predichas	
		True	False
Clases reales	True	VP	FN
	False	FP	VN

Tabla 2.2: Ejemplo Matriz de Confusión.
Fuente: Elaboración propia.

Tomando en consideración este ejemplo, los resultados pueden ser agrupados en 4 conceptos, que se detallan a continuación:

- **Verdadero Positivo (VP):** Corresponden a los datos del set identificados como positivos y son clasificados como positivos por el algoritmo de clasificación.
- **Verdadero Negativo (VN):** Corresponden a los datos del set identificados como negativos y que son clasificados como negativos por el algoritmo de clasificación.
- **False Positivo (FP):** Corresponden a los datos del set identificados como negativos y que son clasificados positivos por el algoritmo de clasificación.
- **Falso Negativo (FN):** Corresponden a los aquellos datos del set identificados positivos y que son clasificados negativos por el algoritmo de clasificación.

Tomando las definiciones anteriores se pueden definir las métricas que permiten analizar los resultados de un algoritmo clasificador. Dentro de las más relevantes se encuentran la Exactitud (Accuracy), Precisión (Precision), Exhaustividad (Recall) y F-Measure (F1) [21]. A continuación se describe la definición y la formula para calcular estos indicadores de desempeño.

Exactidud / Accuracy: Esta métrica representa el porcentaje de datos que son identificados correctamente por el clasificador. Como la ecuación 2.10 muestra, esta medida incluye a todas las clases, por lo tanto corresponde a una métrica que entrega un desempeño general del clasificador. Este indicador puede generar una idea errónea de los resultados si las clases no están balanceadas, ya que no considera la diferencia de proporciones entre las clases. Es por esto que existen otras métricas que permiten analizar el desempeño de una clase en particular como lo son los dos siguientes indicadores.

$$Accuracy = \frac{VP+VN}{VP+VN+FP+FN} \tag{2.10}$$

Precisión / Precision: Este indicador representa a la proporción de los casos de una clase predicha que efectivamente corresponden a esa clase. Esta es una métrica independiente para cada clase del modelo, por lo que refleja cuan certero es el clasificador para detectar elementos de una clase en particular. La ecuación 2.11 y 2.12 muestra este indicador para cada una de las dos clases del ejemplo.

$$Precision_{True} = \frac{VP}{VP+FP} \tag{2.11}$$

$$Precision_{False} = \frac{VN}{VN+FN} \tag{2.12}$$

Exhaustividad / Recall: Esta medida corresponde a la proporción de los casos reales que fueron efectivamente clasificados en sus respectivas clases. Al igual que la *precisión*, este indicador se puede calcular para cada una de las clases. La ecuación 2.13 y 2.14 muestra el calculo de este indicador para cada una de las clases del ejemplo.

$$Recall_{True} = \frac{VP}{VP+FN} \tag{2.13}$$

$$Recall_{False} = \frac{VN}{VN+FP} \tag{2.14}$$

F-Measure / F1: Esta métrica mezcla la *precision* y *recall*, permite analizar la compensación de estas dos medidas. El valor de β es real y positivo y generalmente se le asigna el valor de 1. La ecuación 2.15 define el calculo de este indicador.

$$F-Measure = \frac{(1+\beta) \times Precision \times Recall}{\beta^2 \times (precision + recall)} \tag{2.15}$$

2.6.2. Métricas de Acuerdo

Con el fin de determinar el grado de acuerdo o concordancia que poseen dos observadores a la hora de clasificar un dato, existe una métrica frecuentemente utilizada para evaluar este fenómeno: el coeficiente de *Kappa de Fleiss* [41]. Es una medida estadística que ajusta el efecto del azar en la proporción de la concordancia observada para variables categóricas. Corresponde a una medida más robusta que el simple calculo del porcentaje de acuerdo entre dos observadores. Esta métrica se diferencia de *Kappa de Cohen* [43], la cual solo sirve cuando se evalúa el acuerdo entre no más de dos clases. El calculo para Kappa de Fleiss se muestra en la ecuación 2.16.

$$\kappa = \frac{\mathbb{P}_{acuerdo} - \mathbb{P}_{azar}}{1 - \mathbb{P}_{azar}} \tag{2.16}$$

- $\mathbb{P}_{acuerdo}$ corresponde al acuerdo relativo en los observadores y es equivalente al *accuracy* en una matriz de confusión.
- $\mathbb{P}_{azar}$ representa a la probabilidad hipotética de acuerdo por azar.

El rango de κ se encuentra entre -1 y 1, siendo el valor 0 un acuerdo completamente al azar, mientras que 1 es un acuerdo perfecto.

Para determinar que tan de acuerdo están dos observadores se definen rangos para el valor de κ, los cuales permiten una mejor interpretación de este indicador. En [44] Landis y Koch realizaron una primera caracterización para este valor en 5 rangos diferentes, expuestos en la tabla 4.6.

Rango de *Kappa*	Interpretación
0 - 0.20	Leve
0.21 - 0.40	Razonable
0.41 - 0.60	Moderado
0.61 - 0.80	Considerado
0.81 - 1	Casi perfecto

Tabla 2.3: Interpretación Kappa de Cohen de Landis y Koch.
Fuente: Elaboración propia.

Sin embargo esta interpretación ha sido criticada por otros autores, ya que no existe evidencia que la apoye. Landis y Koch se basaron en su propia intuición personal para determinar los rangos de acuerdo. En [41] indican que esta interpretación es más perjudicial que útil si es que existen varias categorías, por lo que redefinen los rangos y su interpretación para dar una mayor flexibilidad. La tabla 4.7 muestra esta nueva clasificación para el valor de *Kappa de Fleiss*.

Rango de *Kappa*	Interpretación
0 - 0.40	Pobre
0.41 - 0.75	Razonable a Bueno
0.76 - 1	Excelente

Tabla 2.4: Interpretación Kappa de Fleiss.
Fuente: Elaboración propia.

2.6.3. K-Fold Cross Validation

También conocido como validación cruzada en español, es un método utilizado en el proceso de entrenamiento de un clasificador que permite dividir el set de entrenamiento en dos partes, la primera para entrenar el modelo y la segunda para evaluar las métricas de desempeño [45]. El termino *K* hace referencia a la cantidad de sub-sets en que se dividirán los datos. Además este número determina la cantidad de iteraciones que se realizarán, es decir, dado un valor de *K* se efectúan esa misma cantidad de clasificadores y se evalúa cada uno de forma individual. Finalmente las métricas de desempeño son equivalente al promedio simple de los resultados de las *K* iteraciones. La figura 2.8 muestra de manera gráfica el proceso de división del set de datos.

2.7. Modelo de Tópicos

En Machine Learning y procesamiento de lenguaje natural, Topic Modeling [46] es un tipo de modelo estadístico que permite descubrir patrones o "temas" que se encuentran en una colección de documentos. Un "tema" se define como un patrón recurrente de palabras que ocurren conjuntamente.

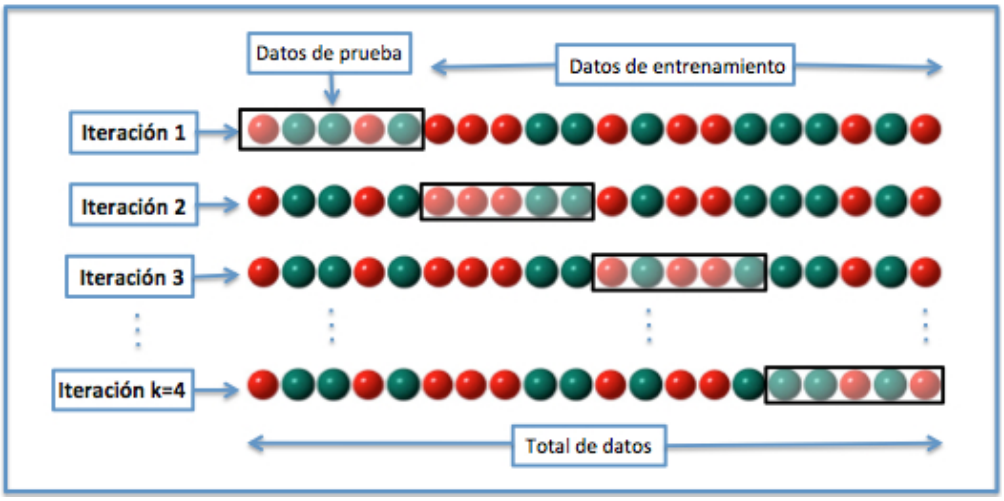

Figura 2.8: Ejemplo de 4-Fold Cross Validation.
Fuente: Wikipedia - Validación cruzada.

El modelado de tópicos es una herramienta de minería de texto de uso frecuente para el descubrimiento de estructuras semánticas ocultas en un texto. Dado que un documento trata de un tema en particular, se podría esperar que algunas palabras aparezcan en el documento más o menos frecuentemente. Esta técnica ha sido utilizada en diversos ámbitos, como por ejemplo categorización de correo electrónico, publicaciones, documentos legales, inclusive tweets [47].

Los "temas"producidos por las técnicas de modelado de tópicos son clusters de palabras similares. Como intuición general, un modelo de tópicos utiliza un marco matemático que permite examinar un conjunto de documentos y descubrir cuales son los tópicos y términos frecuentes en cada uno de ellos condicionando por la probabilidad de pertenencia de las palabras.

A continuación se describe el modelo de tópicos Latent Dirichlet Allocation, que es el más frecuentemente utilizado.

2.7.1. Latent Dirichlet Allocation

Latent Dirichlet Allocation [48] o LDA es un modelo estadístico generativo de tópicos que permite que los conjuntos de observaciones sean explicados por grupos no observados que explican por qué algunas partes de los datos son similares. Por ejemplo, si las observaciones son palabras dentro de documentos, se postula que cada documento es una mezcla de un pequeño número de temas y cada palabra es atribuible a uno de los temas del documento.

La idea detrás de LDA es modelar documentos como si ellos pertenecieran a varios temas, donde un tema se define como una distribución sobre un vocabulario fijo de términos. Específicamente, se supone que los K temas están asociados con una colección y que cada documento presenta estos temas con diferentes proporciones.

En LDA, los datos observados son las palabras de cada documento y las variables ocultas representan la estructura de los tópicos, es decir, los temas y cómo cada documento los exhibe. Dada

una colección de documentos, la distribución posterior de las variables ocultas determina una descomposición de tópicos de la colección.

El proceso generativo de tópicos que lleva a cabo LDA es el siguiente: los documentos se representan como mezclas aleatorias sobre temas latentes, donde cada tema se caracteriza por una distribución sobre las palabras. LDA asume el siguiente proceso generativo para una colección D compuesto de M documentos cad<uno de extensión N_{i}:

1. $\theta_{\mathrm{d}} \sim Dir(\alpha)$, donde $\mathrm{d} \in \{1,...,D\}$
2. $\varphi_k \sim Dir(\beta)$, donde $k \in \{1,...,K\}$
3. Para cada palabra d,n donde $n \in \{1,...,N_{\mathrm{i}}\}, \mathrm{d} \in \{1,...,D\}$
   (a) Escoger un tópico $z_{\mathrm{d},n} \sim \text{Multinomial}(\theta_{\mathrm{i}})$
   (b) Escoger un término $w_{\mathrm{d},n} \sim \text{Multinomial}(\varphi_{z_{\mathrm{d},n}})$

Lo que de forma gráfica se puede ver en la figura 2.9

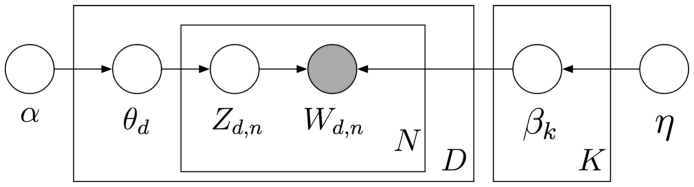

Figura 2.9: Representación gráfica Latent Dirichlet Allocation.
Fuente: Figura 2 en [46].

En LDA se tiene dos parámetros que no son determinados por el modelo, pero de ellos depende gran parte la forma y la separación que los tópicos tengan, estos son los hiper-parámetros α y β de la distribución Dirichlet. Blei [46] indica que el valor de α y β se encuentra directamente relacionado con la dispersión de las probabilidades de los documentos para cada tópico. En la figura 2.10 se pueden observar 15 simulaciones para 10 grupos con distintos valores de α. El eje x son las asignaciones a cada grupo mientras que el eje y corresponde a las probabilidades, es decir con el valor de α es posible controlar que tanto peso se le da a los grupos ya seleccionados cuando se quiere asociar un nuevo elemento a un grupo. En general con un valor alto de α un elemento es asociado a muchos grupos, mientras que con un valor pequeño ocurre el caso contrario.

Griffiths y Steyvers [50] analizaron el comportamiento de LDA en función de los hiper-parámetros α y β, sugieren utilizar un valor de $\alpha = 50/K$, donde K corresponde al número de tópicos, y un valor de $\beta = 200/w$ o 0.1, donde w corresponde al número de palabras en el vocabulario.

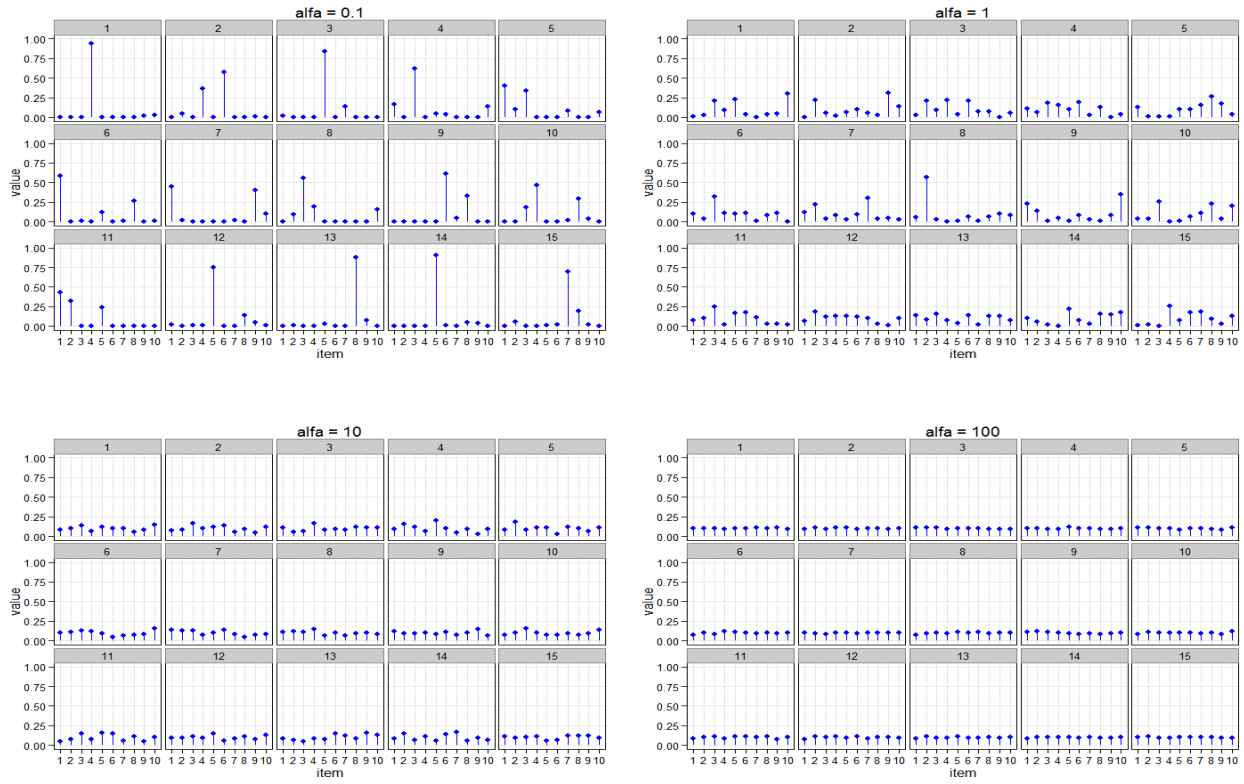

Figura 2.10: Distribución Dirichlet con diversos valores de alfa
Fuente: Figura 2.3 en [49]

2.8. Reclamo

2.8.1. Definición

Un reclamo corresponde a una disconformidad expresada que es manifestada con el fin de que la contraparte tome alguna acción al respecto. Esta disconformidad está relacionada directamente con los productos y servicios adquiridos. El hecho de realizar un reclamo, sin importar el medio, no constituye una denuncia legal y no se inicia un procedimiento para sancionar al proveedor. Lo que se busca a través de un reclamo es que el proveedor se haga responsable y solucione el problema rápidamente y de manera directa con el consumidor.

Por ejemplo, si no se está conforme con el servicio de telefonía que se contrató, el proveedor debe responder siempre y cuando la causa sea razonable y amerite la responsabilidad de este. Para lo cual existen múltiples medios por el cual el cliente puede comunicarse o expresar su descontento para que la empresa se haga cargo y pueda solucionar el problema directamente, tales como:

- Contacto telefónico (Call center)
- Página Web oficial del proveedor
- Redes Sociales (Facebook y Twitter)
- Entidades públicas (SERNAC, Subsecretarias, entre otras)

Es necesario aclarar que un reclamo no es lo mismo que una queja, un reclamo tiene lugar cuando un proveedor ha hecho algún tipo de compromiso con un consumidor y no está cumpliendo con dicho compromiso, es decir, un reclamo es cuando se tiene cierto derecho de exigir algo por lo que se ha pagado y que no se está obteniendo. Por otra parte, una queja es el malestar o descontento por algo que está relacionado indirectamente al producto o servicio adquirido o se refiere a una mala atención de parte de la organización. Por ejemplo, si se acude a un restaurante y el mozo atiende de manera descortés, se puede presentar una queja. También es posible realizar una queja cuando los servicios higiénicos están desaseados. De esta manera, el proveedor podrá implementar mejoras en la atención brindada al cliente. De igual forma que un reclamo, una queja tampoco constituye una denuncia legal, por lo que no se inicia un procedimiento para sancionar al proveedor.

2.8.2. Reclamos en Twitter

En relación a las redes sociales,específicamente en Twitter, las opiniones de reclamo que aquí se exponen poseen un bajo contexto asociado, ya que los mensajes se encuentran acotados a tan solo 140 caracteres de longitud. Lo que genera una dificultad a la hora de identificar si una opinión posee o no carácter de reclamo.

En lo que se refiere a Twitter, las organizaciones hacen esfuerzos para solucionar los reclamos que los usuarios expresan por este medio de comunicación. Para lo cual poseen cuentas exclusivas que se encargan de responder inquietudes, problemas y consultas de los consumidores. Por ejemplo en el rubro de la banca, la empresas Santander Chile tiene dos cuentas: *@SantanderCLnews* , y *@santanderchile*. La primera tiene como objetivo ser el canal oficial de noticias del Banco Santander Chile, mientras que la segunda se encarga de responder las dudas de los clientes.

2.8.3. Reclamos en la Empresa

Un simple episodio de insatisfacción de un cliente puede pasar, en cuestión de horas, a convertirse en un problema que llegue a afectar a las ventas de la compañía o a la cotización de sus acciones. Por ejemplo, la compañía Dell, líder en fabricación de ordenadores personales, sufrió el conocido como “Dell Hell”. Al publicar un blogger insatisfecho una carta abierta al presidente de la compañía, hecho que fue reflejado en todas las revistas y periódicos económicos, y llegó a tener trascendencia tanto en las ventas de ordenadores como en los protocolos y maneras de actuación de la empresa a partir de ese entonces [13].

En términos generales, los reclamos son una consecuencia natural de cualquier actividad de servicio, porque“los errores son una característica inevitable de todos los esfuerzos humanos y por lo tanto de la entrega de un servicio” [51]. El responder y hacerse cargo de los reclamos de los clientes en una organización, se refiere a las estrategias que las empresas utilizan para resolver y aprender de los fallos y problemas que ocurren en los productos y servicios que ofrecen, con el fin de recuperar la confiabilidad frente a sus clientes. Este proceso conlleva una serie de etapas en donde las dos partes, tanto el cliente como el proveedor interactúan para un beneficio mutuo. El cliente se beneficia con la resolución de su problema, mientras que la organización obtiene un consumidor más contento.

Una arista interesante para ver los reclamos es desde la evaluación y reconfiguración de las estrategias en los servicios y productos debido a debilidades en operaciones existentes dentro de una organización. Lo que puede generar el aumento de compromiso y la lealtad de los clientes frente a la empresa si los reclamos son identificados y solucionados oportunamente. Este proceso de resolver los problemas de los consumidores está fuertemente ligado con la satisfacción de los clientes, la confianza y el compromiso. Además un reclamo puede ser visto como la oportunidad de demostrar su integridad para las empresas, ya que les permite identificar situaciones invisibles para ellas, por lo que pueden ser considerados como oportunidades de mejora.

Como dato estadístico, en relación de los clientes que han pasado por un proceso de reclamo, más de la mitad de estos consumidores al terminar este proceso se sienten más negativos frente la empresa después de terminar una situación de reclamo, esto puede ser altamente perjudicial para las empresas dada la alta conectividad y la facilidad para las personas de viralizar situaciones en donde las empresas pasan a llevar a sus clientes. Por lo que hacer frente a los reclamos disminuye el daño del efecto boca en boca que producen los clientes y usuarios insatisfechos y mejora el desempeño final de la organización (rentabilidad de la empresa).

Por el lado de las estrategias de Marketing, los reclamos pueden ser analizados desde dos puntos de vista: Estrategias Ofensivas y Defensivas [52, 53]. El principal objetivo de una estrategia de marketing ofensiva es generar nuevos clientes, mientras que una estrategia de marketing defensivo consiste en retener a los clientes actuales de la empresa. Haciendo la analogía de estas estrategias de marketing con los reclamos, se puede mencionar que las ofensivas hacen relación con quitar clientes de las competencia, esto implica identificar los servicios en que las otras empresas poseen deficiencia. Esto se puede llevar a cabo mediante el análisis en los reclamos que realizan sus clientes, con el objetivo de atraerlos. Por el otro lado, las estrategias de marketing defensivas se enfocan en disminuir la rotación y fuga de los clientes. Considerando que el costo de capturar un nuevo cliente es alto en relación al costo de retener a los clientes propios de una organización, esta estrategia es primordial para las empresas. Esto se ve agudizado en rubros como las telecomunicaciones, más hoy en día en Chile donde la competencia es muy fuerte y no existe prácticamente ningún costo en migrar de una empresa a otra (desde el año 2011 que comenzó la portabilidad en Chile, los clientes se pueden cambiar y no perder su número). En resumen, esta estrategia corresponde a hacerse cargo de los reclamos con el fin de resolverlos y que los clientes se queden en la empresa.

Por último, el propósito del análisis agregado de reclamos es identificar los problemas de los consumidores que ocurren consistentemente a lo largo del tiempo y posiblemente a través de diferentes productos [53]. Esta información agregada se utiliza para formular estrategias de gestión diseñadas para reducir los efectos negativos de los reclamos. El principal beneficio asociado con el análisis agregado de reclamos es que permite a la empresa administrar su negocio de manera proactiva. Mientras que el manejo de reclamos individuales es una parte importante de la gestión de reclamos, se ocupa de los síntomas y no de las causas. El propósito del análisis agregado de reclamos es identificar las causas. Esta información permite a las empresas hacer cambios en el negocio para eliminar o reducir la fuente de los reclamos [54].

Capítulo 3

Modelos de Clasificación de tópicos en Twitter

El estudio sobre la categorización de tweets ha sido motivo de varias investigaciones a nivel mundial. La manera de categorizar opiniones con el fin de entregar información resumida y agregada, como también el permitir filtrar los comentarios ha sido un tema de gran interés desde hace algunos años.

En el siguiente capítulo se mencionan diversos modelos, ordenados por fecha de publicación, que categorizan opiniones de Twitter, para luego ver el que será usado posteriormente.

3.1. Investigación de Bharath, Sriram, et al.

En esta investigación [55] los autores se plantean el desafío de buscar una manera para la clasificación automática tweets distinta al enfoque tradicional de Bag-Of-Words, debido a que postulan que presenta ciertas limitaciones. Los autores logran diseñar una nueva manera de clasificar tweets basada en variables externas a los tweets que permiten clasificar de una mejor manera que el método tradicional.

3.1.1. Set de Datos

El set de datos utilizado en el desarrollo de este modelo fue una colección de tweets de usuarios aleatorios, en donde se eliminaron los tweets que no estuvieran escritos en idioma inglés y que tuvieran menos de 3 palabras, exceptuando URLs y palabras de saludos. El set estaba compuesto por 5.047 tweets correspondientes a 684 autores diferentes. Este set fue etiquetado manualmente en base a 5 categorías predefinidas. Luego de quitar las Stop-Words quedaron 6.747 términos únicos.

3.1.2. Tópicos

Para esta investigación se agruparon los tweets en 5 categorías predefinidas que se puede observar en la tabla 3.1, además se exponen la cantidad de tweets pertenecientes a cada tópico.

Tópico	cantidad
Noticias	2.107
Opiniones	625
Ofertas	1.100
Eventos	1.057
Mensajes privados	518
Total	5.407

Tabla 3.1: Detalle cantidad de tweets por categoría.
Fuente: Elaboración propia.

3.1.3. Procesamiento de Texto

Los autores definen un subconjunto de características del texto para construir un modelo de aprendizaje. Usaron una estrategia codiciosa para escoger las características, las cuales están directamente relacionadas a las clases definidas. Extrajeron 8 variables de los tweets que se describen a continuación:

1. Autor, si corresponde a un periodista, un famoso o un político.
2. Presencia de abreviaciones o jerga.
3. Presencia de frases de tiempo[1].
4. Presencia de palabras de opinión[2].
5. Presencia de énfasis en palabras[3].
6. Presencia de signo de moneda ($) y porcentaje (%).
7. Presencia de @user al comienzo del tweet.
8. Presencia de @user entremedio del tweet

La primera variable es de tipo nominal y las 7 restantes son de carácter binario. Estas variables fueron definidas en base a las características distintivas que los autores reconocieron de los tweets analizados.

[1] se definen como frases que contienen un lugar e información de tiempo (hora)
[2] los autores poseen una lista con cerca de 3.000 palabras de opinión.
[3] se define por la presencia de mayúsculas o el uso de letras repetidas

3.1.4. Modelos y Desempeño

Los experimentos realizados son llevados a cabo mediante la implementación del modelo de clasificación supervisado Naive Bayes gracias a la librería WEKA [56]. Para la evaluación del modelo se utiliza validación cruzada con 5 iteraciones, es decir, separaron el set de datos en 5 partes y en cada iteración utilizaron el 80 % de los datos para entrenar el modelo y el 20 % restante para evaluarlo. En la figura 3.1 se muestra el accuracy general para los 5 modelos analizados por los autores.

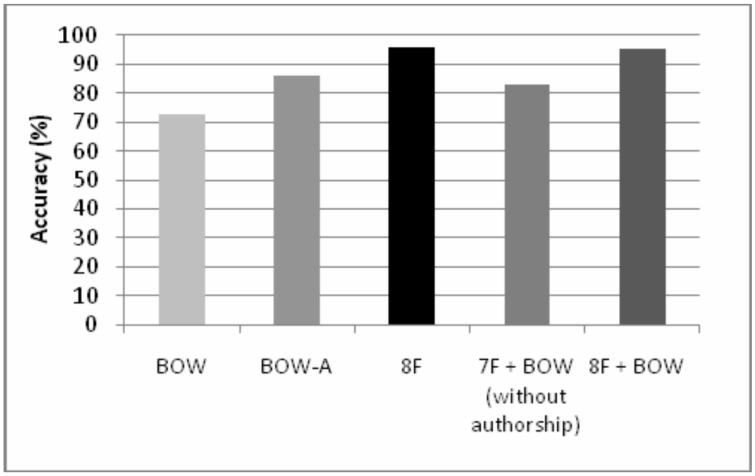

Figura 3.1: Accuracy

A continuación se explica en detalle el significado de cada modelo:

1. **BOW:** representa el enfoque clásico de Bag-Of-Words, el cual presental el peor desempeño de los cinco modelos.

2. **BOW-A:** corresponde a Bag-Of-Words en conjunto con la variable *Autor* definida por los autores.

3. **8F:** Es un modelo de características, que utiliza tan solo las 8 variables definidas y explicadas anteriormente. Este es el modelo que mejor accuracy alcanza llegando al 95 %.

4. **7F+BOW (sin Autor):** corresponde a Bag-Of-Words en conjunto con 7 de las 8 variables definidas.

5. **8F+BOW:** representa las 8 variables definidas en conjunto con Bag-Of-Words. Este modelo presenta un accuracy peor que las 8 variables por si sola.

3.1.5. Conclusiones

Los resultados experimentales muestran que la precisión de la clasificación es alta incluso sin meta-información y el enfoque propuesto supera a la tradicional estrategia de Bag-Of-Words.

Los autores concluyen que la estrategia de Bag-Of-Words se desempeña decentemente, pero su enfoque de 8 variables es significativamente mejor (un 32.1 % de mejora) con este set de categorías.

Además enfatizan que los datos más ruidosos pueden empeorar el rendimiento del enfoque propuesto; por lo tanto, las técnicas de eliminación de ruido son necesarias en estos casos.

Cabe destacar que los resultados de esta investigación están por encima de otras investigaciones realizadas en el mismo tema, sin embargo, tanto el set de datos que utilizaron, como las variables definidas están especialmente diseñadas para el experimento realizado y no son aplicables directamente en otro dominio.

3.2. Investigación de Fernandéz Anta, Antonio, et al.

Esta investigación [57] del año 2013 intenta analizar y comparar distintas aproximaciones de los modelos de detección de tópicos y análisis de sentimiento[4] realizados en el idioma inglés, pero aplicado a un corpus de tweets en español.

3.2.1. Set de Datos

Se utilizó un Corpus de tweets entregados en el contexto del certamen de TASS [58]. Estaba compuesto por 70.000 tweets, pero 7.000 de estos representaban el set de entrenamiento etiquetados con un tópico y una clasificación de sentimiento. Los experimentos fueron realizados con el set de entrenamiento, considerando 5.000 tweets para entrenar el modelo y 2.000 para evaluarlo.

3.2.2. Tópicos

El set de datos contenía tweets correspondientes a 10 categorías distintas: Música, Economía, Entretenimiento, Política, Cine, Literatura, Otros, Tecnología, Deportes y Fútbol.

3.2.3. Procesado del Texto

- **N-Grams :** Encontraron que en la práctica el utilizar un valor de n mayor a 3 no mejoraba los resultados por lo que se consideró usar hasta tri-grams. Definieron remover las palabras que tuviesen una ocurrencia inferior a 5 veces dentro del set de datos de entrenamiento. Además quitaron aquellas palabras que no aparecían en al menos un 85 % en un tópico.
- **Lematización/Stemming :** Utilizaron el software Freeling[5] para realizar el proceso de lematización. En conjunto usaron el software Snowball[6] para realizar el procedimiento de stem-

[4] clasificación de polaridad del texto.
[5] http://nlp.lsi.upc.edu/freeling/node/1
[6] http://snowballstem.org/

ming.

- **Corrección de palabras :** Corrigieron los errores de escritura gracias a la ayuda del diccionario Hunspell [7]. Además otra técnica de preprocesado de texto fue expandir los emoticones[8], abreviaturas y jerga utilizada en mensajes de textos. Observaron que para la detección de tópicos, las palabras determinantes corresponden a los sustantivos y los verbos, por lo que tan solo consideraron estos términos en el set de datos de entrenamiento.
- **Hashtags, Users y URLs :** Utilizaron los hashtags y links contenidos en los tweets para ayudar en la clasificación de tópicos. Para el caso de los url's extrajeron los tags HTML: *h1, h2, h3 y title* de la página a la cual hacían referencia. Además se utilizaron los autores de los tweets, @user.

3.2.4. Modelos y Desempeño

Se utilizó el software WEKA que corresponde a una colección de algoritmos de Machine Learning enfocados en clasificación y clustering. Como métodos de clasificación se evaluaron tan solo cinco: Ibk (k-Nearest Neighbors algorithm), Complement Naive Bayes, Naive Bayes Multinomial, Randon Committee y Sequential minimal optimization.

Se realizaron 14 configuraciones diferentes, combinando los distintos métodos de preprocesado de texto, donde el modelo que mejor desempeño obtuvo fue Complement Naive Bayes con un accuracy de 58,45 %, y su configuración de preprocesado de texto fue: lematización, hashtags y users. La tabla 3.2 muestra el detalle de la precisión, recall y F-Measure y su promedio ponderado para cada una de las 10 categorías del set.

Class	Precision	Recall	F-Measure
Música	0,468	0,619	0,533
Economía	0,316	0,318	0,317
Entretenimiento	0,565	0,503	0,532
Política	0,721	0,814	0,765
Cine	0,386	0,354	0,37
Literatura	0,175	0,241	0,203
Otros	0,551	0,442	0,491
Tecnología	0,194	0,162	0,176
Deportes	0,419	0,5	0,456
Fútbol	0,5	0,409	0,45
Prom. pond.	0,579	0,584	0,578

Tabla 3.2: Detalle Detección de tópicos mediante Complement Naive Bayes.
Fuente: Elaboración propia.

[7] corrector ortográfico utilizado por LibreOffice, OpenOffice.org, Mozilla Firefox 3 & Thunderbird, Google Chrome, y también usado por propietarios de paquetes de software, como Mac OS X, InDesign, memoQ, Opera y SDL Trados. https://hunspell.github.io/

[8] transformar los emoticones en palabras. Ejemplo: :-) en feliz

3.2.5. Conclusiones

Los autores concluyen que ninguno de los métodos analizados es un clasificador excelente para tweets en idioma español. Ninguno de los algoritmos analizados presenta una diferencia significativa con respecto a los demás. Además corroboran que trabajar con textos cortos como los tweets implica un desafío principalmente por la brevedad y la falta de contexto.

3.3. Investigación de Batista, Fernando, et al.

Esta investigación [59] del año 2013 presenta una estrategia basada en clasificadores binarios de máxima entropía para el análisis de sentimiento y categorización de textos de Twitter enfocados al idioma español. El modelo desarrollado consigue los mejores resultados para la categorización temática.

3.3.1. Set de Datos

Los experimentos desarrollados en esta investigación corresponden a datos de Twitter en español, entregados en el contexto del certamen de TASS [58]. El set de datos contenía 7200 tweets etiquetados con la polaridad y su respectivo tópico. Los autores consideraron un 80 % del set para entrenar el modelo (5.755 tweets) y el restante 20 % lo utilizaron para evaluar el desempeño de este (1.445 tweets).

Los tweets además contenían información acerca de del usuario quien emitió dicho tweet. Esta información se refiere a si el usuario corresponde a un periodista, un famoso o un político.

3.3.2. Tópicos

El set de datos contenía tweets correspondientes a 10 categorías distintas: Música, Economía, Entretenimiento, Política, Cine, Literatura, Otros, Tecnología, Deportes y Fútbol.

3.3.3. Procesado de Texto

En primera instancia realizaron una tokenización de los tweets utilizando *twokenize*[9].

- **Marcas de Puntuación**
- **Negación**

  Las palabras que estén precedidas por una negación como "nunca.º "no", se crea una nueva palabra con el prefijo "NO_"seguido de la palabra.

[9]Librería con métodos y funciones para realizar tokenización. https://github.com/brendano/ark-tweet-nlp

- **Transformación de URLs**

  Los links presentes en los tweets fueron transformados al token "HTTP".
- **Hashtags**

  Para los hashtags dentro de los tweet fue creada una nueva palabra que consiste en eliminar el prefijo # de esta. El hashtag original también se mantiene.
- **Caracteres Repetidos**

  Si un tweet contiene palabras que tengan al menos 3 letras iguales consecutivas se genera una nueva token "LONG_WORD", y a su vez se eliminan los caracteres repetidos de la palabra original.
- **Mayusculas**

  Las palabras en mayúsculas se mantienen, pero a su vez se genera la misma palabra en minúsculas.
- **USER**

  Se crea una nueva variable para cada tweet correspondiente al nombre de usuario quien emite el mensaje.
- **Tipo de Usuario**

  Se incorpora el campo de tipo de usuario definido en el corpus entregado, que corresponde a uno de los tres tipos descritos.
- **bi-grams**

  Se consideran la utilización de bi-grams, pero solo en las siguiente combinaciones: "HTTP", hashtags (sin #), USER, LONG_WORD y las palabras en minúsculas. Esta combinación consiste en combinar tuplas de tokens de este conjunto y formar el bi-gram correspondiente.

3.3.4. Modelos y Desempeño

Como algoritmo de clasificación se basaron en el método de regresión logística multinomial, el cual corresponde al modelo de clasificación de máxima entropía para eventos independientes. Los autores justifican el uso de este método por la características del problema, es decir, por la dispersión de la matriz resultante del modelo Bag-Of-Words. En la tabla 3.3 se muestra el detalle por categoría de las métricas de desempeño del modelo desarrollado por los autores.

3.3.5. Conclusiones

Los autores concluyen que el modelo diseñado obtuvo los mejores resultados en el contexto del certamen TASS, además, destacan que en su modelo la variable “tipo de autor” no muestra una contribución significativa en la capacidad predictiva del modelo, mas bien, se muestra bajo sus expectativas iniciales.

Mencionan que para mejorar el desempeño es posible utilizar la polaridad del texto (la conformidad de la opinión emitida en el tweet), en conjunto con información relacionada al usuario (por ejemplo: el número de tweets, seguidores y amigos). Además señalan que utilizar un lexicón o jerga atingente al corpus analizado puede generar un gran impacto en el desempeño final.

Tópico	Precision	Recall	F-Measure
Política	0.89	0.87	0.88
Literatura	0.48	0.44	0.46
Música	0.90	0.47	0.62
Deportes	0.52	0.63	0.57
Tecnología	0.71	0.63	0.67
Cine	0.70	0.44	0.54
Fútbol	0.54	0.65	0.59
Entretenimiento	0.93	0.48	0.63
Economía	0.87	0.47	0.61
Otros	0.64	0.91	0.75
Promedio.	0.79	0.77	0.78

Tabla 3.3: Detalle Detección de tópicos mediante Maximum Entropy.
Fuente: Elaboración propia.

3.4. Investigación de Ebert, Sebastian, et al.

Esta investigación [10] del año 2013 trata de diseñar un método de detección automático de reclamos para una compañía aseguradora de Alemania. Se exponen los desafíos y la complejidad para abordar este problema y se analizan dos aproximaciones. En primer lugar se utiliza el método estándar de Bag-Of-Words y como segunda técnica se analiza el enfoque basado en características del texto de NLP.

Los autores sugieren que la tarea de detectar texto con temática de reclamos es similar al análisis de sentimientos. Suponen que los documentos con carácter de reclamo están escritos de una manera negativa.

3.4.1. Set de Datos

Para los experimentos realizados se utilizaron 5 sets de datos distintos, cuatro de estos corresponden a documentos de correspondencia recibidos por departamentos de una empresa aseguradora alemana, y el otro set corresponde a un corpus de análisis de películas de IMDb[10] el cual se encuentra etiquetado con la polaridad de cada texto. En la tabla 3.4 se muestra el detalle de los tamaños para cada corpus utilizado en los experimentos.

Para los cuatro corpus de documentos de correspondencia de la aseguradora, el proceso de etiquetado fue realizado por un empleado de la empresa. Sin embargo para el set de datos de IMDb, el proceso de etiquetado que desarrollaron los autores fue definir como *Reclamo* aquellos con polaridad positiva y como *No Reclamo* aquellos con polaridad negativa (esta decisión tuvo como razón

[10]Base de datos de películas en Internet, es una base de datos en linea que almacena información relacionada con películas, personal de equipo de producción (incluyendo directores y productores), actores, series de televisión, programas de televisión, videojuegos, actores de doblaje y, más recientemente, personajes ficticios que aparecen en los medios de entretenimiento visual. http://www.imdb.com

la simpleza, ya que los textos ya se encontraban etiquetados con su respectiva polaridad)

Corpus	Reclamos	No Reclamos	Nº palabras
Liability	55	170	6.039
Car	1.088	2.610	66.961
Damage 1	373	989	34.674
Damage 2	372	865	31.461
IMDb	1.000	1.000	38.911

Tabla 3.4: Detalle Set de Datos Complaint Detection.
Fuente: Elaboración propia.

3.4.2. Tópicos

Esta investigación busca diseñar un método de detección de reclamos, por lo que las categorías definidas son tan solo dos, si el documento presenta carácter de reclamo o no.

3.4.3. Procesado de Texto

Los documentos son representados mediante el modelo de Bag-of-Words, además utilizaron la notación SMART [60] para las ponderaciones cada una de las palabras. Consideraron 18 configuraciones diferentes para las ponderaciones de los términos.

Realizaron tres técnicas de preprocesado de texto, las cuales se detallan a continuación:

- **Stemming**

  Para realizar esta tarea usaron el algoritmo de stemming de la librería Snowball en idioma alemán para los 4 corpus de la aseguradora y para el set de datos de películas lo realizaron con el algoritmo de stemming en inglés.
- **Remover StopWords**

  Para este proceso utilizaron el set de stopwords que provee el proyecto Snowball. Se utilizaron 174 y 231 stopwords para los idiomas alemán e inglés respectivamente.
- **Análisis de componentes principales**

  Otra técnica utilizada tuvo como objetivo el reducir la cantidad de características. PCA (Principal Component Analysis) corresponde a una método no supervisado para transformar una matriz de grandes dimensiones en otra con menos.
- **N-Grams**

  Finalmente el último paso que realizaron fue generar n-grams para incorporar contexto. Se decidieron por usar bi-grams y tri-grams.

3.4.4. Modelos y Desempeño

Como algoritmos de clasificación los autores utilizaron Support Vector Machine, en su implementación LibSVM, y para obtener las métricas de desempeño del modelo, realizaron validación cruzada con 10 iteraciones, es decir, ejecutaron 10 veces el algoritmo dejando un 90 % de los datos para entrenar y un 10 % de estos para evaluarlo. Y el resultado final corresponde al promedio de las 10 iteraciones. Los indicadores de desempeño que midieron fueron: Precision, recall y F-Measure para la clase de *Reclamos*.

Luego de analizar las 18 configuraciones para los pesos de los términos, encontraron que la mejor configuración para los corpus previstos es considerando tan solo la presencia o la ausencia de las palabras en el documento. Además la remoción de stop-words y stemming tuvo un impacto positivo en tan solo dos de los cinco corpus, y los autores mencionan que las diferencias no son estadísticamente significativas, por lo que no recomiendan ninguna de las dos técnicas de preprocesado.

Por el lado de PCA, los autores dan el visto bueno a esta técnica, mencionan que reducir el número de variables, manteniendo un 95 % de la varianza de los datos, el desempeño del modelo se mantiene. Redujeron las variables en más de un 96 % mediante PCA.

Finalmente en el caso de n-grams, los autores señalan que no se logra capturar el contexto mediante esta técnica.

En la tabla 3.5 se muestra el desempeño del modelo propuesto por los autores para cada uno de los 5 corpus analizados.

	Liability		Car		Damage 1		Damage 2		IMDb	
	P	F1	P	F1	P	F1	P	F1	P	F1
bxx	0.83	0.75	0.81	0.78	0.75	0.69	0.87	0.86	0.85	0.85
Stemming	0.80	0.73	0.80	0.77	0.75	0.69	0.88	0.86	0.85	0.85
StopWords	0.94	0.70	0.79	0.75	0.77	0.68	0.87	0.82	0.87	0.86
PCA	0.84	0.75	0.79	0.76	0.74	0.68	0.85	0.84	0.85	0.84
bi-grams	0.84	0.67	0.84	0.79	0.84	0.69	0.88	0.84	0.88	0.87
tri-grams	0.83	0.51	0.84	0.77	0.84	0.65	0.89	0.82	0.89	0.88

Tabla 3.5: Detalle Performance Complaint Detection.
Fuente: Elaboración propia.

3.4.5. Conclusiones

Los autores concluyen que la representación binaria de las palabras mostró los mejores resultados, además de tener el beneficio de ser rápido computacionalmente. El incorporar contexto a través de n-grams, la utilización de stemming y stopwords no fue beneficioso; por el contrario PCA es una buena técnica. Finalmente mencionan que para mejorar los resultados se necesita mayor conocimiento lingüístico del texto, como por ejemplo la polaridad del texto.

3.5. Resumen y Elección

Como primera observación de las principales investigaciones que se han realizado hasta fecha en relación a la detección de tópicos en Twitter es que el enfoque clásico de Bag-Of-Words es suficientemente bueno para esta tarea, sin embargo, cuando se incorpora información adicional acerca de los tweets, por ejemplo datos del autor, signos de puntuación, keywords, entre otros; el desempeño del clasificador mejora.

En segundo lugar, para el desafío de detectar opiniones de reclamo en Twitter, el trabajo realizado en [10] da buenas luces de que es posible construir un clasificador, aunque existen diferencias evidentes en los tipos de documentos de la investigación y los tweets, principalmente por el largo de los textos y la falta de contexto que presentan los comentarios en Twitter. Sin embargo, el hecho de que el enfoque de Bag-Of-Words haya entregado buenos resultados es alentador para comprobar la hipótesis propuesta.

En tercer lugar, dentro de los algoritmos de clasificación supervisada que se han utilizado en las investigaciones no hay certeza en que uno es mejor que otro. El algoritmo a utilizar depende esencialmente de la configuración que se utilice, es decir, depende del set de datos, de las clases definidas, de las técnicas de preprocesado utilizadas, etc. Sin embargo, algoritmos como Naive Bayes y Support Vector Machines tienden a entregar mejores resultados que otros, siendo que estos son los más rápidos (en recursos y tiempo computacional) para entrenar. Por lo cual estos dos algoritmos serán los primeros a evaluar.

En cuarto lugar, se observar que en general se utiliza 5 fold cross validation, esto corresponde a segmentar los datos en un 80 % para entrenar el modelo, dejando el 20 % restante para validarlo y obtener las métricas de desempeño. Esto permite disminuir el sobre-ajuste que pueda llegar a presentar el modelo construido.

En quinto lugar, el modelo que se busca construir debe tener la particularidad de poder predecir datos no antes visto, más aún, datos que aun no han sido generados. Por ende se pretende abarcar un intervalo de tiempo lo más amplio posible para disminuir al mínimo el sesgo que puedan presentar los datos para entrenar el modelo.

Finalmente, se toma en consideración evaluar tan solo el enfoque de Bag-Of-Words, debido a que no se tiene información adicional al texto, como por ejemplo tipos de autor del mensaje, u otra información que permita diferenciar si una opinion es o no un reclamo. Por lo que un enfoque en base a características no es factible de realizar para este trabajo. Se pretende aplicar el modelo clásico BOW aplicando las mejores prácticas de preprocesado de texto vistas en estas investigaciones como: incorporar marcas de puntuación, eliminar sufijos de hashtags, caracteres repetidos, bi-grams, URL's, stemming, remoción de Stopwords, etc.

Capítulo 4

Diseño y Construcción del set de datos

El presente capítulo tiene como objetivo definir los requerimientos, el diseño de y la posterior construcción del juego o set de datos a utilizar para el clasificador de opiniones de reclamos en Twitter.

Como punto inicial en la construcción del set de datos se deben definir los elementos que lo conformarán, en otras palabras esto implica acotar el universo de Twitter a un segmento con características similares, con el fin de desarrollar un clasificador que posea una precisión acorde a las investigaciones realizadas en clasificación de tópicos en Twitter. En lo que sigue, se explicará el proceso de selección de un rubro relevante para el desarrollo de los algoritmos de clasificación.

4.1. Elección de Rubro

Es importante mencionar que el abarcar a todo el espectro de industrias o rubros en la construcción de un único clasificador es ambicioso y tal vez imposible. Por lo que es necesario acotar el universo a un segmento pequeño y definido de empresas que presenten características similares para lograr una precisión semejante a las investigaciones analizadas en el capítulo anterior.

Dentro de los puntos a considerar que sirven como fundamento para escoger un rubro en específico por sobre todo el universo de Twitter se encuentran que:

1. **Contexto y forma de expresión:** Las empresas se desenvuelven en un contexto específico, y por ende la forma de expresarse de las personas hacia las empresas es distinta. El tipo de lenguaje utilizado para comunicarse es diferente entre rubro y empresas, por ejemplo en algunas empresas el lenguaje utilizado es menos formal, como lo es el caso de Virgin Mobile, por lo cual no se puede incorporar todo dentro un mismo clasificador. A continuación se puede apreciar las diferencias que existen en el lenguaje y jerga que utilizan los usuarios para comunicarse con las empresas en Twitter:

    - **Virgin Mobile:**
      “Cumpitas y a los que ya abiamos contratado un antiplan no nos regalan el extra? xd

@VirginMobile_cl”

- **Santander Chile:**

  “@santanderchile llevo toda la mañana intentado hacer una transferencia y comunicarme con su callcenter y no puedo hacer ninguna! Que ocurre?”

2. **Lenguaje utilizado:** Las personas en Twitter se expresan y reclaman usando términos que se inscriben en un contexto determinado. El contexto es un punto crucial en la detección de un reclamo, ya que palabras utilizadas en un contexto pueden significar algo completamente distinto en otro. Por ejemplo, los siguiente tweets utilizan el término “cuenta”, pero tiene un significado completamente distinto en ambos. Para Movistar significa el pago del servicio, mientras que para Santander representa el servicio de cuenta para almacenar dinero.

   - **Movistar:**

     “@AyudaMovistarCL llevo más de 30 días sin poder llamar. Les falló el sistema. Me cobraron cuenta completa y ahora cortaron servicio. Un asco”

   - **Santander Chile:**

     “Ojo los que tengan cuenta en @santanderchile. Segunda vez en el mes que me llega este correo falso.”

3. **Categorías:** Las categorías en las cuales se inscriben las opiniones son innumerables. Además, estas dependen específicamente del rubro el cual se esté analizando, mas aún pueden existir empresas las cuales posean tan solo una parte de las categorías de un rubro. Por ejemplo en el rubro de las telecomunicaciones existen 5 servicios definidos y en el rubro financiero existen 12, y estos no tienen ninguna relación entre ellos.

Tomando estos puntos, es que se debe construir un clasificador particular para cada rubro, y tal vez para cada empresa de ser necesario, de modo de poder capturar todas las características en donde se desenvuelve la empresa y así construir un clasificador que entregue información útil.

Dicho lo anterior, se debe acotar el alcance del módulo de reclamos a desarrollar a solo un rubro, con el objetivo de validar la hipótesis propuesta. Para determinar cual es el mejor rubro se analizaron los que reciben la mayor cantidad de reclamos en Chile y se escogió el más sobresaliente desde el punto de vista de los reclamos, tanto en general como en Twitter. Para lograr esto se analizaron las industrias desde distintos aspectos:

1. La cantidad de reclamos que recibe el rubro, con el fin de poder escoger una industria que sea relevante, para que el clasificador a desarrollar agregue valor en la detección de reclamos.

2. La cantidad de empresas que conforman al rubro y su concentración de mercado, con el fin de identificar con anterioridad dificultades que se puedan presentar al momento de desarrollar el clasificador, como por ejemplo que existan muchas empresas y por tanto haya una alta variabilidad en el contexto y tipo de reclamos que se presenten en dicha industria.

3. La cantidad de productos y/o servicios que se ofrecen en la industria, con el fin de medir la cantidad de clases a priori en que deberá categorizar el clasificador a diseñar.

4. El comportamiento de las empresas del rubro en la red social de Twitter, ya sea si poseen presencia, si disponen de cuentas especializadas para el contacto por los consumidores, la

cantidad de mensajes que reciben, entre otros.

Con el fin de estudiar los reclamos que acontecen en Chile, se analizó a la organización encargada de manejar los reclamos de los usuarios y clientes hacia las empresas: el SERNAC[1]. Esta entidad periódicamente publica informes acerca del comportamiento de los reclamos en Chile, en donde expone la cantidad de reclamos recibidos a través del tiempo, realiza una segmentación en base a productos y servicios, entre otros. Dentro de los informes públicos que emite el SERNAC, el último estudio realizado sobre el estado de las industrias con respecto a los reclamos [61] fue en el año 2014, donde analizaron los reclamos que llegan a los distintos rubros durante los años 2012 y 2013. Este estudio reveló que existen tres rubros que reciben una cantidad significativa de reclamos en comparación a los demás, estos son Telecomunicaciones, Servicios Financieros y Locales Comerciales. En promedio, de los años analizados, el rubro de las Telecomunicaciones es el que más reclamos recibió. En la figura 4.1 se muestra el plano general sobre la cantidad de reclamos recibidos por el SERNAC a los 24 rubros identificados por esta institución.

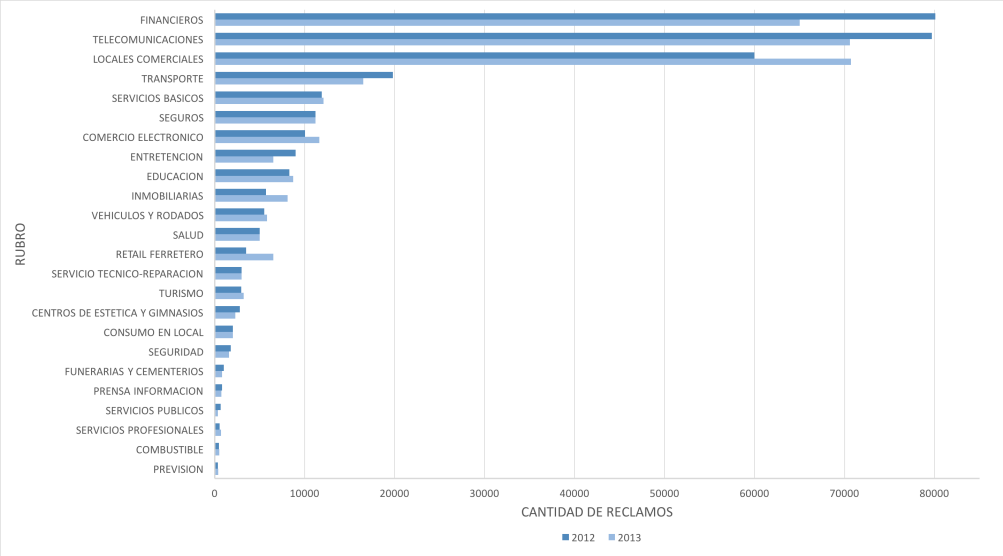

Figura 4.1: Reclamos ante SERNAC por rubro en los años 2012 y 2013
Imagen adaptada de [61].

En base a esto, los tres rubros antes mencionados son los candidatos a ser parte del set de datos, ya que concentran la mayor cantidad de reclamos. En lo que sigue se analizarán estos rubros con el fin de aclarar sus principales diferencias, las que serán claves para la elección del rubro a considerar para el set de tweets.

4.1.1. Industria Financiera

En primer lugar, el mercado financiero es quien más reclamos recibió durante el año 2012, además en otro informe emitido por el SERNAC [62], que trata sobre los reclamos destinados al esta industria, mostró que el Banco Estado es la empresa que posee el mayor volumen de reclamos durante el primer trimestre del año 2014 y 2015, seguido del Banco Santander Chile y el Banco de

[1]Servicio Nacional del Consumidor - Ministerio de Economía, Fomento y Turismo. http://www.sernac.cl/

Chile respectivamente. Además estas últimas dos empresas concentran al 60 % de los deudores de la banca nacional, considerando que este mercado existe un total de 15 empresas. En la tabla 4.1 se puede apreciar la cantidad de reclamos de este rubro en los cinco productos más reclamados. Además de estos 5 productos que se muestran, existen otros 7 servicios que poseen una menor cantidad de reclamos y en la tabla quedan agrupados en "Otro": Cobranza extrajudicial (6.2 %), Cajero automático (4.9 %), giros (3.3 %), línea de crédito (2.4 %), crédito educacional aval del estado (2.3 %), tarjeta de débito (2.1 %), cuenta de ahorro (1.7 %) y otros (5.0 %).

BANCO	Créd. cons.	T. crédito	Cta. Cte	Créd. hipotec.	Cta. vista	Otro	Total
Estado	295	340	269	300	661	1130	2.995
Santander	473	285	297	264	111	505	1.935
De Chile	371	355	220	81	86	311	1.424
BCI	225	155	171	95	95	263	1.004
BBVA	168	122	150	145	52	153	790
Falabella	182	136	146	21	47	202	734
Scotiabank	97	59	75	124	15	167	537
Paris	308	102	0	4	0	63	477
Corpbanca	113	47	54	51	12	87	364
Ripley	79	30	0	5	0	49	163
Itau	30	21	35	23	8	26	143
Security	10	17	60	8	7	14	116
Consorcio	20	8	6	23	8	19	84
BICE	2	1	6	5	4	7	25
Internacional	3	0	0	1	0	2	6
Total	2.376	1.678	1.489	1.150	1.106	2.998	10.797

Tabla 4.1: Distribución de los reclamos por banco, según producto financiero.
Fuente: Tabla adaptada de [62].

A partir de esto, se puede extraer que en esta industria existe una alta concentración de mercado en unas pocas empresas, así también, los productos y servicios se encuentran muy bien definidos y todas las empresas los proveen, lo cual entrega buenos indicios para utilizar esta industria en el desarrollo del clasificador. Ahora queda analizar esta industria en la red social de Twitter, ya que este corresponde al punto en donde se centrará la investigación.

Analizando a estas tres empresas que concentran la mayor cantidad de reclamos a través del SERNAC, en la red social de Twitter se pueden observar características importantes a considerar sobre ellas. A continuación se detalla el estado de cada una de estas empresas en Twitter.

1. **Banco Estado:** Solo posee una cuenta en Twitter: *@BancoEstado*, la cual la definen como el *"Canal oficial en Twitter"*. En esta cuenta concentran tanto las consultas, inquietudes y reclamos que realizan sus clientes, como también la información que el banco publica periódicamente, como promociones, nuevos productos y servicios, publicidad, etc. Esto genera una complicación, ya que en una misma cuenta se agrupa mucha información. En términos de la cantidad de mensajes que recibe esta cuenta, se observó que durante la segunda mitad del mes de Septiembre de 2016 a esta cuenta llegaron en promedio 40 a 50 mensajes por día.

2. **Banco Santander:** Esta empresa posee dos cuentas en Twitter: @*SantanderCLnews* y @*santanderchile*. La primera es una cuenta definida como el *"canal oficial de noticias de Banco Santander Chile"*, mientras que la segunda es una cuenta exclusiva de ayuda y soporte para sus clientes, en donde se encargan de resolver dudas y consultas de ellos. En relación a la cantidad de tweets que llegan a esta cuenta de ayuda y soporte, durante la segunda mitad del mes de Septiembre de 2016 recibieron entre 30 a 40 mensajes por día.

3. **Banco de Chile:** Este banco al igual que el anterior posee dos cuentas en Twitter: @*bancodechile* y @*AyudaBancoChile*. La primera es la la cuenta oficial donde publican beneficios, descuentos y promociones, mientras que la segunda es la cuenta oficial de Ayuda que se encarga de responder dudas, consultas y reclamos. En relación a la cantidad de tweets que llegan a esta cuenta de ayuda y soporte, durante la segunda mitad del mes de Septiembre de 2016 recibieron entre 10 a 20 mensajes por día.

Evaluando la panorámica de estas tres empresas en el ámbito de Twitter, se puede observar que existe una correlación entre la cantidad de reclamos que las empresas reciben a través del SERNAC con los mensajes recibidos en sus cuentas de ayuda y soporte de Twitter. Sin embargo, la cantidad de tweets parece ser baja, tomando en cuenta que para la construcción de un clasificador se necesita una gran cantidad de datos, y que comprendan un periodo significativo (idealmente desde que la creación de las cuenta en Twitter).

Es necesario mencionar que en este rubro de la banca existe una característica fundamental, la cual puede condicionar la baja cantidad de mensajes de reclamos e inquietudes presentes en sus cuentas de Twitter. Esta es la presencia de ejecutivos de cuentas designados para los clientes, que se encarga de ser el principal medio de contacto con la empresa. Por medio de ellos los clientes pueden dar a conocer y gestionar sus cuentas, además de revolver dudas y problemas que presenten con los productos y servicios que tengan contratados. Dicho todo lo anterior, este rubro no parece ser una buena opción para ser considera en el desarrollo de este trabajo para validar la hipótesis propuesta.

4.1.2. Industria de Locales Comerciales

En tercer lugar de las empresas con reclamos a través del SERNAC, se encuentra el rubro de los locales comerciales. Dentro de este rubro la cantidad de empresas es enorme, contiene desde las grandes empresas de retail como: Falabella, Ripley, Paris; hasta locales pequeños y exclusivos que se enfocan en productos y categorías especificas. Además, las grandes empresas nombradas anteriormente son transversales a múltiples rubros, ellas presentan productos y servicios en rubro de la banca, seguros, entre otros, por lo que no es posible acotarlas a tan solo una industria. Siguiendo esta línea, la cantidad de categorías presentes en este rubro es inmensa, basta con ingresar a alguna página web de una de estas empresas para ver la gran cantidad de categorías y subcategorías que existen. Considerando esto, lo mas recomendable sería incluir a tan solo las grandes empresas de retail que abarcan todos los segmentos de categorías y además son bastante similar entre ellas, lo que dejaría de lado por el momento a las tiendas exclusivas de menor tamaño.

Habiendo entonces decidido contemplar solo a estas tres grandes empresas de retail, se puede continuar con el análisis previsto. Observando la clasificación que ellas mismas han definido en

sus página web, se pueden encontrar entre 11 a 14 categorías distintas, donde las más relevantes son: Electro, Tecnología, Electro-hogar, Muebles, Dormitorio, Deporte, Infantil, Belleza, Moda, Calzado y Otros. Dicho esto, es factible combinar a estas tres empresa, ya que las categorías son idénticas en todas ellas.

Ahora es necesario evaluar el estado de las empresas en Twitter.

1. **Ripley:** En primer lugar la empresa Ripley posee tan solo una cuenta en Twitter: *@Ripley-Chile*, que tiene la función de comunicar nuevos productos, promociones, publicidad, etc. No tienen el foco de responder consultas ni reclamos de los clientes. Además a esta cuenta llegan dudas e inquietudes de los clientes, pero muy puntuales, 1 o 2 por día. Por lo cual se puede concluir que esta empresa no tiene una presencia activa en Twitter con respecto a la atención con sus clientes.

2. **Paris:** En segundo lugar se encuentra la empresa Paris, la cual posee dos cuentas: *@ClientesParis* y *@tiendas_paris*. La primera es la encargada de ser el canal oficial de noticias y marketing de la empresa para dar a conocer lo nuevo, mientras que la segunda es la dedicada a la atención de clientes, sin embargo, tiene una muy baja actividad, menos de 3.500 tweets en su historia y durante la segunda mitad del mes de Septiembre de 2016 recibieron alrededor de 5 mensajes por día, por lo que se puede considerar que no posee una presencia activa en esta plataforma.

3. **Falabella:** Finalmente la empresa Falabella, posee dos cuentas en Twitter: *@Falabella_Chile* y *@FalabellaAyuda*. La primera es la cuenta oficial de la compañía y tiene el fin de ser el medio de marketing en la red social. La otra cuenta se dedica a la atención de clientes para responder reclamos y dudas de ellos. Esta es la empresa que posee la mayor actividad de las tres analizadas en la red social de Twitter, durante la segunda mitad del mes de septiembre de 2016 recibieron en promedio 30 a 40 tweets por día.

En conclusión este rubro aparentaba ser muy bueno desde el punto de vista de que son solo tres empresas a considerar y que las categorías se encuentran muy bien definidas y son idénticas para todas las empresas. Sin embargo, en la red social de Twitter, que es donde se enmarca el desarrollo del trabajo, no existen suficientes datos para construir un modelo clasificador, por lo que esta industria se descarta para ser la considerada a validar la hipótesis propuesta.

4.1.3. Industria de Telecomunicaciones

Finalmente se encuentra el rubro de las telecomunicaciones, el cual posee la mayor cantidad de reclamos a través del SERNAC durante el año 2013. En otro de los documentos públicos que emitió el SERNAC [63], se puede observar que los servicios ofrecidos en esta industria son cinco:

- Telefonía Móvil
- Internet Móvil
- Telefonía Fija
- Internet Fijo
- Televisión Paga

En Chile existen tan solo 17 empresas que proveen alguno de estos servicios. De todas estas empresas en el mercado, existen cuatro de ellas que concentran la mayor participación del mercado en cada uno de los 5 servicios que se ofrecen en este rubro. En mayor detalle estas cuatro empresas concentran más del 75% del mercado para cada uno de los servicios mencionados anteriormente. La tabla 4.2 detalla la participación de mercado de todas las empresas según serivio. Estas cuatro empresas corresponden a: Movistar, Entel, VTR y Claro.

Empresa	Servicio				
	Tel. móvil	Inet. móvil	Tel. fija	Inet. fija	Televisión
Movistar	38,18%	38,98%	44,76%	38,24%	21,49%
Entel	35,61%	31,74%	14,93%	0,80%	2,98%
Claro	23,02%	25,05%	7,50%	11,82%	15,78%
WOM	1,15%	1,39%	-	-	-
Virgin	1,01%	1,26%	-	-	-
VTR	0,55%	1,05%	20,81%	37,66%	34,97%
Falabella	0,41%	0,42%	-	-	-
Telsur	0,04%	0,07%	4,77%	6,39%	3,05%
Simple	0,02%	0,03%	-	-	-
Netline	0,01%	0,02%	0,39%	0,03%	-
Colo-Colo	0,01%	0,00%	-	-	-
GTD	-	-	5,50%	2,21%	0,87%
CTR	-	-	0,71%	0,34%	-
CMET	-	-	0,63%	0,63%	0,39%
Mundo Pacífico	-	-	-	1,88%	2,80%
Tuver	-	-	-	-	0,29%
Directv	-	-	-	-	17,39%
Total Abonados	22.974.998	10.525.635	3.434.579	2.596.253	2.912.356

Tabla 4.2: Participación de Mercado Telecomunicaciones según servicio - primer trimestre 2015. Fuente: Elaboración propia.

De la tabla 4.3, que analiza la cantidad de reclamos que llegan a este rubro en el primer trimestre de 2015, se puede observar que existe un correlación entre la cantidad de abonados y la cantidad de reclamos.

Teniendo entonces que existen unas pocas empresas que son relevantes en esta industria, sigue por analizar el estado de estas en la red social de Twitter.

1. **Movistar:** Esta empresa posee dos cuentas en Twitter: *@MovistarChile* y *@AyudaMovistarCL*. La primera se encarga de entregar novedades, beneficios, concursos y más, mientras que las segunda representa el canal oficial de ayuda y soporte de la empresa. Con respecto a la cantidad de tweets que recibe diariamente la cuenta encargada de soporte con sus clientes, se tiene que durante la segunda mitad del mes de Septiembre de 2016 esta cantidad fue superior a 200 mensajes al día.

2. **Entel:** Esta empresa posee cuatro cuentas en la red social de Twitter: *@entel*, *@entel_ayuda*, *@zonaEntel* y *@entel_empresas*. La primera está dedicada a entregar información referente

a marketing, promociones, publicidad, etc. La segunda es el canal de ayuda en Twitter de la empresa, donde responder dudas, reclamos, entre otras. La tercera cuenta se encarga solo de entregar información con relación a promociones, concursos, y beneficios del club *ZonaEntel*[2]. La última cuenta se dedica a resolver dudas y problemas de los clientes empresas, la cual posee poca actividad debido a que los clientes empresas representan un pequeña porción de los abonados totales. Tomando entonces en consideración solo la cuenta enfocada en ayuda y soporte a clientes personas, se puede observar que durante la segunda mitad del mes de Septiembre de 2016 recibió aproximadamente 120 mensajes por día.

3. **VTR:** Esta empresa posee dos cuentas en Twitter: *@VTRChile* y *@VTRsoporte*. La primera es la cuenta oficial que se enfoca en entregar información relacionada a publicidad, promociones, concursos, nuevos productos, etc; mientras que la segunda se ocupa de la atención a los clientes por este medio en la resolución de los problemas y dudas que presenten. Durante la segunda mitad del mes Septiembre de 2016 a esta cuenta de ayuda y soporte llegaron en promedio 200 mensajes de parte de sus consumidores.

4. **Claro:** Finalmente esta empresa también considera dos cuentas en Twitter: *@clarochile_cl* y *@miclaro_cl*. La primera se define como el canal oficial donde informan de las actividades y publicidad de empresas, mientras que la segunda tiene el objetivo de ser el medio de contacto ante problema e inquietudes de los clientes en esta red social. A esta cuenta de soporte de Claro durante la segunda mitad del mes de Septiembre de 2016 llegaron alrededor de 50 tweets por día.

A partir del análisis sobre estas cuatro empresas en sus cuenta de atención a clientes en Twitter, la empresa Movistar es quien recibe la mayor cantidad de mensajes a su *@AyudaMovistarCL*t. La figura 4.2 muestra la cantidad de tweets[3] que llegan a la cuenta de soporte de la empresa Movistar en el periodo del 15 de Septiembre el 04 de Octubre de 2016. A simple vista se puede observar que existe estacionalidad con respecto al día de la semana, los valles corresponden a fines de semana, mientras que los picos representan a los días lunes, martes, miércoles y jueves. Aproximadamente a la cuenta de soporte de Movistar en Twitter llegan 250 tweets en días hábiles y 150 en fines de semana.

4.1.4. Resumen y Elección del Rubro a Utilizar

En conclusión luego de este análisis realizado en relación a los 3 rubros que concentran la mayor cantidad de reclamos a través del SERNAC, el rubro de las telecomunicaciones es el escogido para continuar con el desarrollo en la construcción del set de datos, debido a las siguientes tres razones:

- Contiene pocas empresas que concentran gran parte del mercado, por lo que es posible considerar a pocas empresas en el set de entrenamiento para abarcar a gran parte del mercado.
- Contiene productos y servicios muy bien definidos y que son idénticos para todas las empresas de esta industria, lo que permite generar un set de categorías transversal a todas las organizaciones para segmentar las opiniones de reclamos de los clientes.

[2]Club de beneficios de Entel. https://zona.entel.cl

[3]corresponden a mensajes que no son respuesta a otro, no son retweets, ni emitidos por la misma empresa. Fueron extraídos gracias a la REST API con el método Search(Keyword).

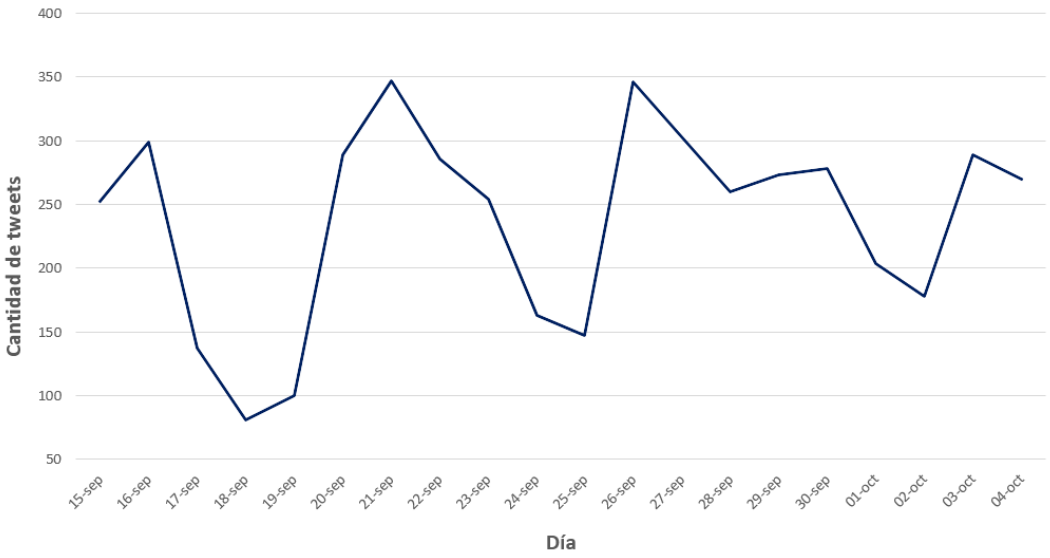

Figura 4.2: Frecuencia de tweets a la cuenta @AyudaMovistarCL entre 15 Sep. al 04 Oct. Fuente: Elaboración propia.

- Existen suficientes datos que los usuarios generan en Twitter en relación a reclamos, consultas, dudas, etc; Además en esta industria la gran mayoría de las empresas disponen de una cuenta exclusiva para la atención de clientes lo que permite acotar aún más el contexto de los tweets, que en general en esta red social es muy escaso.

4.2. Elección de Empresas

Luego de haber escogido el rubro en el cual enfocarse para desarrollar el modelo de clasificación de reclamos, es necesario ahora definir cuales serán las empresas a incluir para el clasificador, en otras palabras, se debe indagar en mayor profundidad este rubro y analizar a las empresas de manera individual en relación a los reclamos que se generan en Twitter.

Como se mencionó en la sección anterior, en Chile el rubro de las Telecomunicaciones contempla cinco servicios entregados por 17 empresas distintas. El criterio de elección de las empresas que formarán parte del set de entrenamiento será escoger a las más representativas, y para lograr esto se analizará el comportamiento de las empresas en:

1. **Participación de mercado**, ya sea en general o por producto dentro de la industria, con el fin de escoger a empresas relevantes.

2. **Cantidad de mensajes** de reclamos, consultas e inquietudes, que los clientes manifiestan a sus proveedores, con el objetivo de construir una clasificador que tengo datos de donde entrenar y a su vez genere valor para la empresa en cuestión.

3. **Cuentas en Twitter**, si poseen cuentas especializadas para responder reclamos y dudas, o si mantienen toda la información en una sola cuenta. Esta característica con el fin de determinar el contexto de los tweets, que permitirá comprender los datos de una mejor manera.

De manera similar al SERNAC, en Chile existe una institución encargada de analizar, regular y gestionar específicamente al rubro de las telecomunicaciones, esta es la Subsecretaría de Telecomunicaciones - SUBTEL. Este organismo lleva un registro constante de todos los movimientos que ocurren en este rubro tales como: la cantidad de suscriptores de cada empresa por servicio, los reclamos que realizan los clientes, la movilidad de clientes, entre otros.

Periódicamente la SUBTEL realiza informes y rankings en donde muestra el estado de las empresas con respecto al mercado y a sus clientes. El último documento publicado [63] comprende al periodo que abarca al primer trimestre del año 2015. Dentro de los distintos temas que trata el documento, se expone el volumen de clientes (o abonados) para cada una de las 17 empresas de telecomunicación presentes en Chile, según los cinco servicios que existen en este rubro.

La tabla 4.2 muestra la participación de mercado para cada una de las empresas en los 5 servicios de las telecomunicaciones, correspondiente al primer trimestre de 2015. Las empresas que no proveen alguno de los servicios se muestran con un guión (-). De esta tabla se puede observar que la empresa con mayor participación en todos los servicios, excepto en Televisión que se encuentra en segundo lugar, es la empresa Movistar. Además considerando a las empresas que proveen todos los servicios, se pueden distinguir fácilmente a las cuatro más grandes:

- Movistar
- Entel
- Claro
- VTR

Cabe destacar que en el servicio de Televisión paga la empresa *DirectTV* se encuentra en tercer lugar en participación de mercado, pero al poseer tan solo un servicio, no califica para conformar el el set de datos, ya que sus reclamos se encuentran contenidos solo a este servicio. En fin, considerando tan solo estas cuatro empresas antes listadas, la participación que ellas alcanzan a nivel global en cada servicio es:

- **Telefonía móvil:** 97,33 %
- **Internet móvil:** 96,82 %
- **Telefonía fija:** 88 %
- **Internet fija:** 88,52 %
- **Televisión:** 75,22 %

Habiendo analizado e identificado a las empresas mas grandes en volumen de clientes, se concluye que en base al primer criterio definido, las empresas *Movistar, VTR, Entel* y *Claro* son las elegidas, ahora sigue examinar a las empresas de la industria de telecomunicaciones con respecto a sus reclamos, con el fin de corroborar si se mantiene esta tendencia. Para esto, el mismo informe [63] presenta la cantidad de reclamos que llegan al SERNAC y SUBTEL, la tabla 4.3 detalla la cantidad de reclamos, segmentados por servicio, recibidos por estas instituciones en el periodo que comprende al primer semestre del año 2015.

Es importante comentar que existen empresas en este rubro que tienen cobertura en regiones específicas del país, como es el caso de TELSUR, que entrega sus servicios solamente entre la Región del Biobío y Los Ríos.

Empresa	Servicio				
	Tel. móvil	Inet. móvil	Tel. fija	Inet. fija	Televisión
Movistar	1.579	164	707	496	238
Entel	1.145	117	97	37	23
Claro	1.269	79	149	104	165
WOM	21	7	-	-	-
Virgin	44	1	-	-	-
VTR	115	42	229	313	41
Falabella	7	0	-	-	-
Telsur	1	1	15	10	4
Simple	2	1	-	-	-
Netline	5	0	0	1	-
Colo-Colo	0	0	-	-	-
GTD	-	-	15	21	6
CTR	-	-	4	6	-
CMET	-	-	6	7	6
Mundo Pacífico	-	-	-	1	4
Tuver	-	-	-	-	8
Directv	-	-	-	-	101
TOTAL	4.187	412	1.222	997	596

Tabla 4.3: Cantidad de reclamos SERNAC y SUBTEL por servicio - Primer trimestre 2015. Fuente: Elaboración propia.

De la tabla 4.3 se puede observar que las mismas cuatro empresas identificadas anteriormente, las que poseen el mayor volumen de clientes, también presentan la mayor cantidad de reclamos a través del SERNAC y SUBTEL, por que a primera vista la tendencia se mantiene, por lo que estas mismas empresas son las elegidas por medio del segundo criterio definido.

Finalmente, tan solo queda estudiar a las empresas en el contexto de la red social Twitter, que será finalmente la fuente desde donde se obtendrán los datos, y por ende este es el punto más importante a considerar para la decisión final de las empresas a incorporar en el set de datos.

Tal como se mencionó en la sección anterior, algunas empresas en Twitter poseen cuentas exclusivas en atención de clientes, cuya finalidad es resolver problemas y dudas de los clientes. A continuación se listan las empresas y sus respectivas cuentas de Twitter dedicadas (si es que poseen) a la atención de clientes.

De la tabla 4.4 se puede extraer que la empresa Movistar y VTR son las que mayor cantidad de

[4]Cuenta general, no es especializada en atención de clientes. https://twitter.com/MovilFalabella
[5]Cuenta general, no es especializada en atención de clientes. https://twitter.com/Telsur
[6]Cuenta suspendida. https://twitter.com/colocolomovil
[7]Sin actividad desde Dic. 2012. https://twitter.com/gtd_manquehue
[8]Sin actividad desde Oct. 2011. https://twitter.com/cmetchile
[9]Sin actividad desde May. 2011. https://twitter.com/cmetchile
[10]Cuenta general, no es especializada en atención de clientes. https://twitter.com/DIRECTVChile
[11]Canal de soporte de América Latina.

Empresa	Cuenta	Prom. Tweets x día
Movistar	@AyudaMovistarCL	200
Entel	@Entel_ayuda	120
Claro	@MiClaro_cl	50
WOM	@WOMteAyuda	35
Virgin	@AloVMCL	5
VTR	@VTRSoporte	200
Falabella	@MovilFalabella[4]	0
Telsur	@Telsur[5]	5
Simple	@Ayuda_Simple	2
Netline	-	-
Colo-Colo	@ColoColoMovil[6]	-
GTD	@gtd_manquehue[7]	-
CTR	-	-
CMET	@CmetChile[8]	-
Mundo Pacífico	-	-
Tuves	@cl_TuvesHD[9]	-
Directv	@DIRECTVChile[10] @DIRECTVServicio[11]	50

Tabla 4.4: Cuentas de Twitter de Empresas Telecomunicaciones Chile - Segunda mitad de Septiembre 2016

Fuente: Elaboración propia.

mensajes reciben en su cuenta de atención a clientes en Twitter, seguido de Entel y Claro respectivamente. Dentro de las otras empresas existentes, ocurre que varias de estas: no poseen cuentas en Twitter, sus cuentas se encuentran inactivas, no separan la atención de clientes del marketing o poseen pocos mensajes; por lo que no califican para ser parte de las empresas a escoger.

4.2.1. Resumen y Elección de Empresas a Utilizar

En conclusión, luego de analizar a todas las empresas del rubro de telecomunicaciones en base a los tres criterios definidos en un comienzo, se decide por escoger finalmente a cuatro empresas: Movistar, Entel, Claro y VTR, debido a las siguientes razones:

1. Estas cuatro empresas entregan todos los servicios identificados en la industria de telecomunicaciones en Chile, por lo que las categorías a clasificar los tweets son transversales y no representan un inconveniente, ya que se busca desarrollar un clasificador que abarque varias empresas.

2. Poseen la mayor cantidad de participación de mercado dentro de la industria, como ya se mencionó estas empresas concentran más del 75 % en cada uno de los cinco servicios, además tienen presencia en todas la zonas del país, característica que otras empresas no disponen.

3. En la red social de Twitter, que es donde se enfoca el trabajo de título, dichas empresas

disponen de cuentas exclusivas para la atención de sus clientes, lo que permite acotar el contexto de los mensajes que reciben, y por ende construir un mejor algoritmo.

4. El punto anterior no tendría ningún valor si es que estas cuentas no tuviesen tweets de sus clientes, particularidad que para estas empresas si ocurre, y son las que más interacción tienen dentro de todas las empresas de este rubro.

4.3. Dinámica en Twitter

En las cuentas de atención a clientes o de ayuda y soporte de las empresas de telecomunicación ocurre una situación muy particular en relación a las conversaciones que se generan, la mayoría de estas siguen un patrón establecido, el cual corresponde a un protocolo por parte de los encargados de gestionar dichas cuentas de Twitter. A grandes rasgos, los pasos que se siguen en este procedimiento se enumeran a continuación:

1. Un cliente envía un mensaje a la cuenta de la empresa, ya sea como una *mención* o *mensaje directo*, buscando una respuesta por parte de la empresa.

2. En una primera etapa, luego de recibido un mensaje por parte de un usuario, se le solicitan datos como rut del titular de la cuenta, número asociado u otro identificador, con el fin de corroborar realizar una correspondencia con los datos de la empresa y corroborar que efectivamente es un cliente.

3. Luego de recibidos y verificados los datos del cliente se procede con:

   (a) Derivar con un ejecutivo especialista, quien analizará el caso y se contactará con el cliente, para resolver su problema.

   (b) Derivar a un call center especialista para resolver la inquietud.

   (c) Dirigirse a la página web la empresa para obtener información referente a su inquietud.

   (d) Entregarle una respuesta (si es que la persona encargada de la cuenta de Twitter posee la facultad y conocimiento) con respecto a la inquietud que presente el cliente.

   (e) Entre otros.

4. Finalmente se termina la conversación.

La importancia de comprender el flujo y la arquitectura de una conversación bajo este contexto de atención de clientes en Twitter, tiene el objetivo de identificar cuales son los tweets que interesan finalmente a incorporar en el set de entrenamiento, ya que una conversación contempla múltiples mensajes y se debe enmarcar la conversación completa dentro de una categoría, no cada mensaje de manera individual. De esta forma interesan solo los tweets que llegan en una primera etapa a estas cuentas, en la figura 4.3 se expone una conversación entre un cliente y la cuenta de ayuda y soporte de Movistar, a modo de visualizar de mejor manera lo planteado. El tweet a considerar es el que se encuentra resaltado con rojo, es decir, aquellos tweets que son los iniciadores en las conversaciones

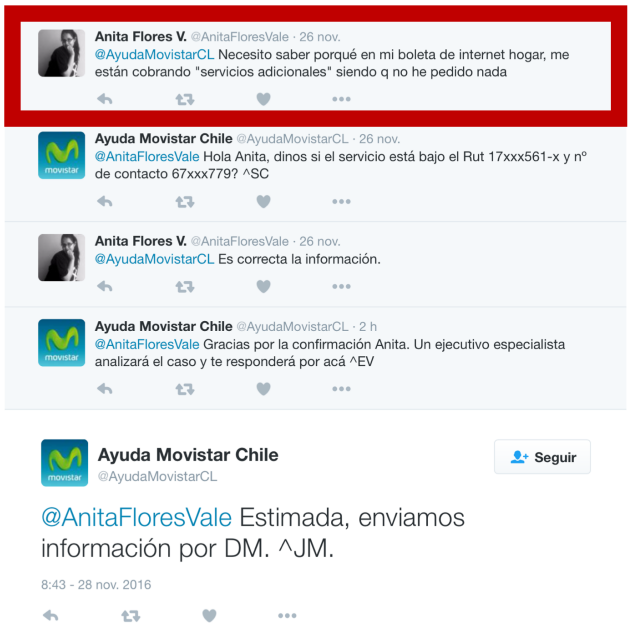

Figura 4.3: Conversación Twitter cuenta @AyudaMovistarCL.

de Twitter, debido a que los otros mensajes de respuesta presentan un contexto distinto.

4.4. Selección y Extracción de Datos

Como se mencionó en un comienzo, el objetivo del módulo de reclamos es detectar las opiniones de reclamo presentes en los mensajes que publican los clientes en las cuentas de ayuda y soporte en Twitter, y luego clasificarlas en categorías predefinidas que tengan relación con los servicios presentes en el rubro de las telecomunicaciones, y otros temas atingentes a los reclamos que se presentan en este. La figura 4.3 muestra de manera gráfica un ejemplo de una conversación que se produce en las cuentas de ayuda y soporte, y como se justificó anteriormente solo se deben considerar los tweets que representan el inicio de una conversación, ya que es en esta instancia donde se explicita la inquietud del cliente, y los mensajes que le siguen presentan un contexto totalmente diferente que se escapan del alcance del modelo a desarrollar.

Dicho esto, no se deben considerar todos los mensajes que llegan a las cuentas de las empresas, solo se deben tener en cuenta los tweets que representan el inicio de una conversación, para lograr esto se requieren filtrar los mensajes en base a características particulares que se detallan a continuación:

- Deben ser menciones o mensajes directos.
- No deben ser respuesta de otro tweet.
- Deben ser emitidos por un usuario de Twitter distinto a la cuenta que se quiere analizar.
- No deben Retweets.
- Deben estar escritos en idioma español.

Estas características indican que (1) deben ser tweets que llegan a estas cuentas en Twitter, (2) el tweet debe ser el primero en una conversación (independiente de otros), es decir, debe ser el primer mensaje que le llega a una cuenta acerca de una consulta, reclamo, etc. (3) El tweet no debe ser emitido por el mismo usuario que se está analizando, ya que en muchas ocasiones el usuario pone su nombre de cuenta en el mensaje que escribe, por lo que estos tweets no deben ser considerados. (5) el tweet no debe ser un Retweet, y finalmente (4) el tweet debe estar escrito en idioma español. El punto número 4 solo aplica para el set de entrenamiento, ya que el modulo final implementado en OpinionZoom debe ser capaz de clasificar y categorizar también los Retweets.

Habiendo definido entonces las características que deben poseer los mensajes a considerar, se debe determinar el método de extracción de tweets, el cual se realiza gracias a la REST API de Twitter. Esta API posee varias funciones, pero las utilizadas para extraer los tweets fueron dos: *Search(Keyword)* y *getUserTimeline(UserId)*. La primera permite extraer tweets que contengan una "keyword" específica, mientras que la segunda permite extraer los tweets que un usuario en especifico ha publicado.

Se utilizaron tres enfoques distintos para la extracción de tweets, pero todos siguiendo una metodología estándar en la selección de las características del tweet. A continuación se listan los pasos realizados para filtrar los tweets que se buscan:

1. En una primera etapa se buscan todos los tweets que contengan alguna de las *keywords* correspondientes a las cuatro cuentas que se buscan analizar.

2. Se filtra por la variable *getInReplyToStatusId() = -1*, esto indica que el tweet es independiente de otro, es decir, no está relacionado con otros en forma de respuesta. Haciendo la correspondencia con lo definido antes, en base a estos se identifican a los tweets que inician una conversación.

3. Se filtra por la variable *getUser().getScreenName() != keyword*, la cual indica el usuario emisor del tweet, el cual debe ser distinto al emisor de la cuenta analizada.

4. Se filtra por la variable *IsRetweet() = false*, la cual indica que el tweet no representa un Retweet.

5. Se filtra por la variable *getLang() = ES*, la cual indica que el tweet está escrito en idioma español.

La cuarta condición antes descrita, se impone como una manera rápida y fácil para no considerar tweets repetidos, ya que los retweets presentan exactamente el mismo contenido de otro tweet emitido.

En lo que sigue se detallarán los 3 métodos ejecutados en la extracción de tweets con sus respectivos resultados.

4.4.1. REST API - Search Keyword

Como primer enfoque para extraer los tweets se utilizó el método *"GET search/tweets"* implementado en la librería Twitter4J, que permite rescatar tweets recientes[12] que contengan una *keyword*. Para este caso se ejecutaron 4 búsquedas, una por cada cuenta investigada.

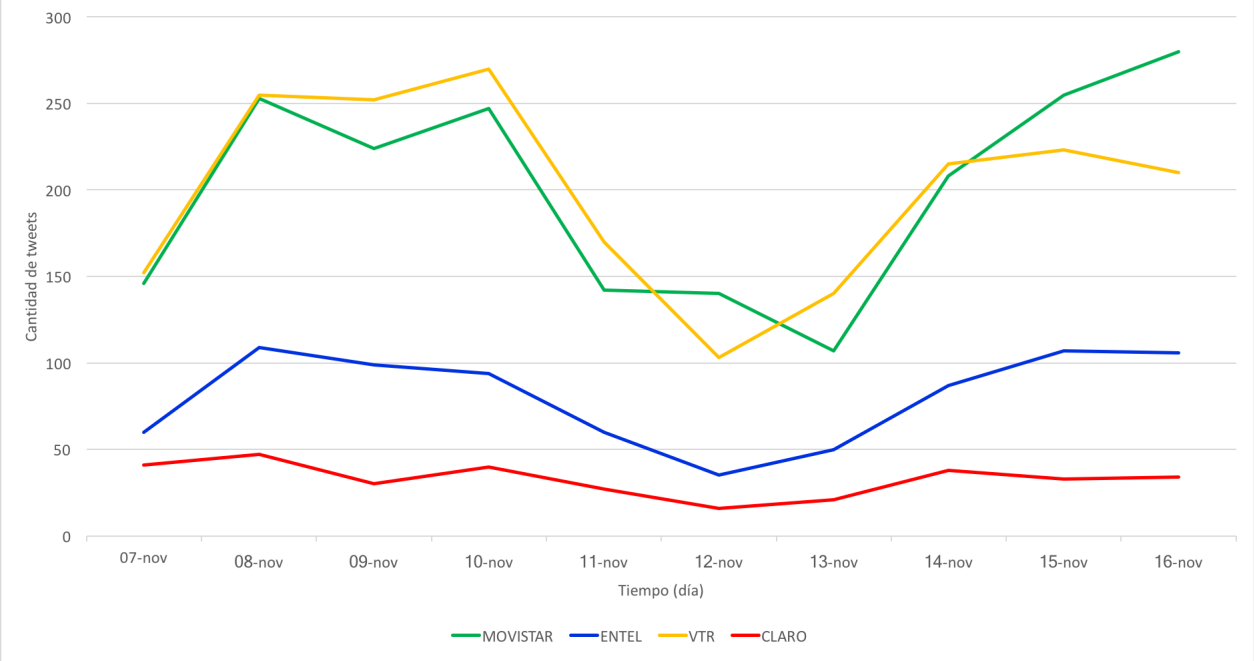

Figura 4.4: Resumen extracción de tweets - REST API Search/tweets.
Fuente: Elaboración propia.

De este proceso se logró extraer un total de 5.126 tweets correspondientes al periodo comprendido entre los días 7 y 16 de Noviembre de 2016. En la figura 4.4 se puede apreciar la cantidad de de tweets extraídos por día para cada una de las cuatro empresas a incorporar en el set de datos.

Es importante destacar que la gran ventaja de este método es que permite rescatar todos los tweets que han sido emitidos durante el periodo que entrega el método *Search* de la REST API. Sin embargo, la desventaja es que solo permite extraer tweets de un lapsus de tiempo muy breve en comparación con la historia de las cuentas de Twitter, 2 semanas v/s 5 años en el caso de Entel y VTR.

4.4.2. Base de Datos *La Gorda*

Dado que el método descrito antes no entrega un gran volumen de tweets, más aun el intervalo de tiempo de los datos corresponde a tan solo 10 días, puede generar que la construcción del set de datos con tan solo estos datos quede sesgado con respecto a algún suceso especial como fallas masivas de algún sistema, o el contexto de ese periodo en particular.

[12]Hasta un máximo de dos semanas de antigüedad.

Con el fin de evitar un sesgo en los datos de procede a buscar datos en la base de datos *La Gorda*, (Base de Datos de OpinionZoom, en donde se almacenan los tweets emitidos por usuarios chilenos[13]).

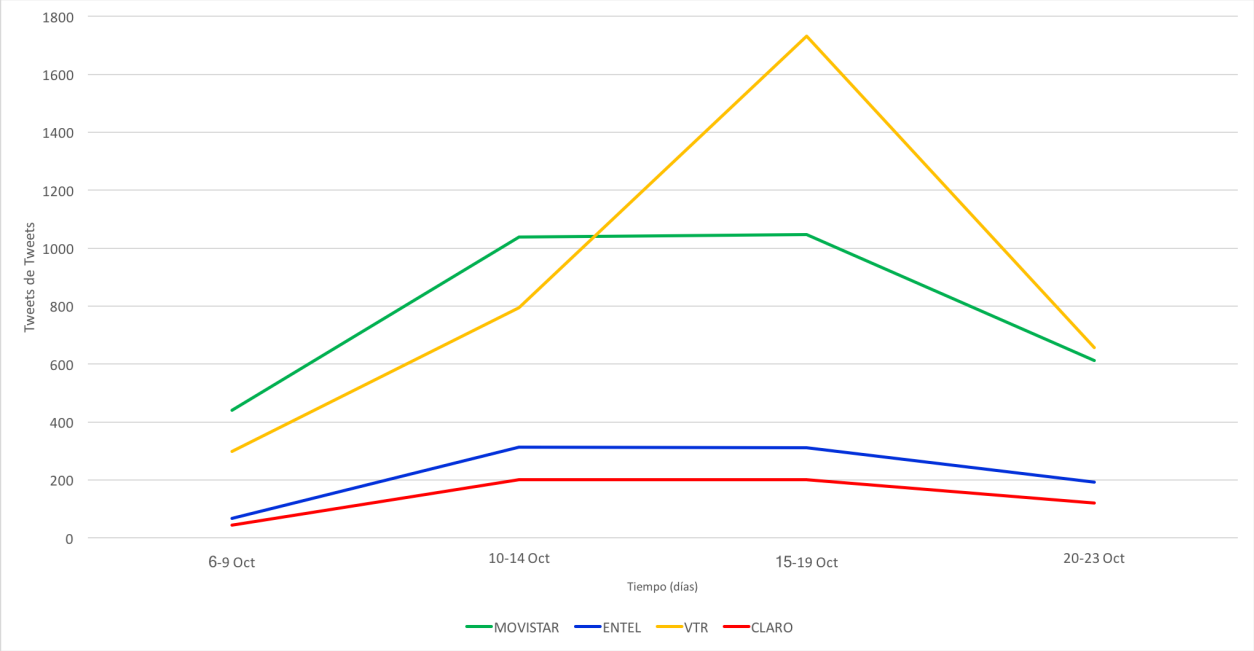

Figura 4.5: Resumen extracción de tweets - Base de Datos La Gorda.
Fuente: Elaboración propia.

Por medio de este método, se logró extraer un set de 8.071 tweets correspondientes al periodo del 6 al 23 de Octubre de 2016. En la figura 4.5 se puede apreciar la cantidad de de tweets extraídos para cada una de las cuatro empresas.

Analizando los datos de los dos primeros métodos de extracción de tweets se pudo observar un hecho bastante interesante con respecto al sistema que utiliza OpinionZoom para recolectar tweets emitidos por los usuarios de Chile. Comparando los resultados de ambos métodos se puede observar que en promedio la base de datos La Gorda posee tan solo un 70 % de los tweets que se pueden obtener por medio de la *Search API* (la cual entrega el 100 % de los tweets en el periodo consultado), cuya causalidad podría corresponder a que no se están siguiendo a todos los usuarios chilenos en Twitter.

En conclusión este segundo método de extracción de tweets tampoco logra cumplir con los requisitos, ya que al igual que el primero, abarca un periodo de tiempo breve. Además presenta la desventaja de no poseer la totalidad de los mensajes emitidos en dicho periodo. Por lo que se ejecutó un tercer método de extracción con el fin de abarcar un rango de tiempo mayor y que los datos no presentaran sesgo con respecto al momento en el cual se emitieron.

[13]Corresponde a una lista de 922.000 usuarios chilenos desarrollada en [64]

4.4.3. Crawler de Usuarios Chilenos

Este último método desarrollado está basado en la implementación realizada en [64], con pequeñas modificaciones para mejorar el tiempo de extracción de los datos.

Este método se basa en revisar los *Statuses* (mensajes emitidos) de un set de 922.220 usuarios chilenos. Gracias a las modificaciones en el código se logró revisar a todos los usuarios en un tiempo menor a 15 horas. Cabe destacar que este modelo presenta ciertas limitaciones, siendo la principal que tan solo se puedan recuperar un máximo de 3.200 tweets por usuario, lo que implica que si un usuario de Twitter ha emitido una cantidad de tweets mayor a este número, tan solo podrán rescatarse los últimos 3.200. Esto ocasiona que no se puedan recuperar todos los tweets históricos, pero es un muy buena aproximación. Mientras más antiguos sean los tweets, existe un menor probabilidad de extraerlos con este método.

En total se lograron extraer 261.856 tweets correspondientes a las cuatro empresas buscadas. La figura 4.6 muestra el resumen de los tweets extraídos mediante este tercer método. Se puede observar que las empresas Movistar y VTR poseen la mayor cantidad de tweets recibidos, mientras que Entel se ha mantenido constante durante los últimos 5 años, y Claro presenta la menor cantidad de tweets recibidos, que se puede deber a la corta vida de esta cuenta. En la misma figura se puede apreciar el inicio de las cuentas en Twitter: Entel y VTR se crearon en el año 2010, Movistar en 2013 y Claro en 2015.

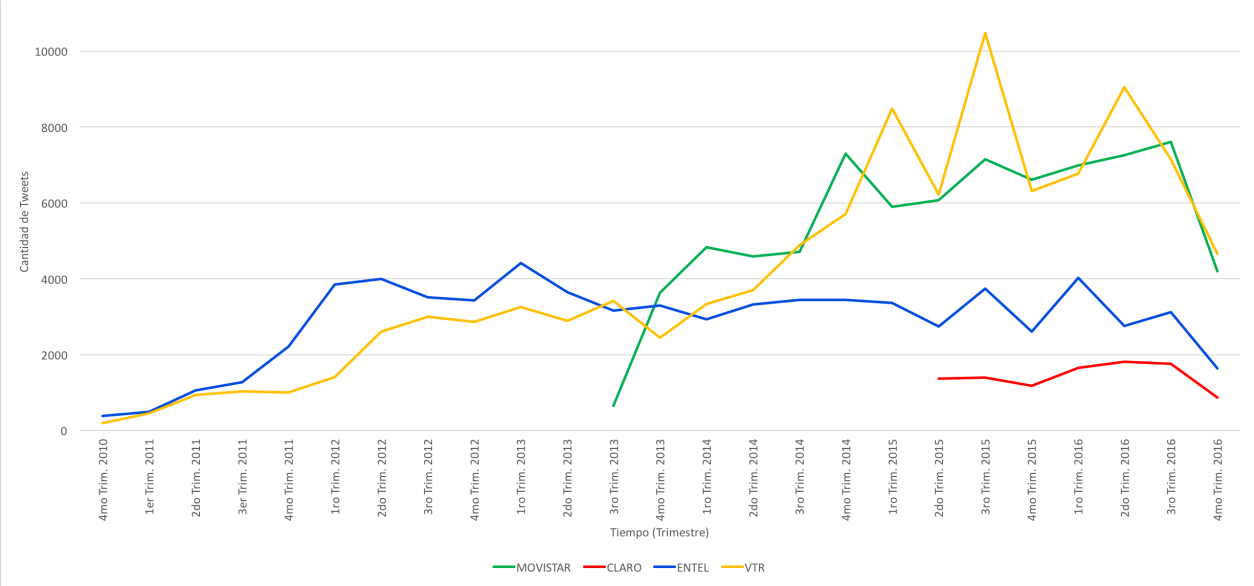

Figura 4.6: Resumen extracción de tweets - Crawler usuarios chilenos.
Fuente: Elaboración propia.

Este tercer método es el que permite rescatar la mayor cantidad de datos y tiene la capacidad de extraerlos desde los comienzos de las cuentas en Twitter. Sin embargo al igual que en el caso de la base de datos *La Gorda* posee la desventaja que no se revisan a todos los usuarios chilenos, tan solo a un aproximado del 70 % de ellos.

4.4.4. Resumen del Proceso de Extracción de Datos

En resumen, del proceso de extracción de tweets desarrollado se logró rescatar un set de 274.948 tweets que abarcan todo la historia de las cuentas de ayuda y soporte analizadas (desde su fecha de creación en Twitter). Además los datos más recientes representan de mejor forma la realidad, ya que corresponden a una mayor proporción de los datos reales generados en Twitter.

4.5. Tamaño y Selección del Set de Entrenamiento

Primero que todo es necesario volver a destacar que los tweets necesarios para la construcción del modelo de clasificación son los mensajes que los usuarios envían a las cuentas de ayuda y soporte de las empresas en Twitter. Este número no se encuentra explícito en ningún lugar en la página de Twitter, por lo que es necesario realizar una estimación para calcular el universo de mensajes enviados a las cuentas de ayuda y soporte. Tal como se mencionó en la sección anterior, el tercer método de extracción de tweets no permite extraer todo el universo de datos, por lo que el numero real es aún más grande que el encontrado.

Analizando el total de tweets que se muestra en la página de Twitter de las empresas escogidas en Twitter, se tiene que la empresa con mayor cantidad de mensajes emitidos es Movistar, seguida por Entel, VTR y Claro respectivamente. La tabla 4.5 muestra en detalle la cantidad total de tweets emitidos al día 16 de Noviembre de 2016 en las cuentas de ayuda y soporte para cada una de estas empresas.

Empresa	Cuenta de Twitter	Cantidad de tweets (miles)
Movistar	@AyudaMovistarCL	438
Entel	@Entel_ayuda	306
Claro	@MiClaro_cl	56
VTR	@VTRSoporte	150
TOTAL	-	905

Tabla 4.5: Cantidad total de tweets a cuentas ayuda y soporte en Twitter al 16 de Noviembre de 2016.

Fuente: Elaboración propia.

Para el caso que interesa, este número no es el que se necesita analizar, pero bajo el supuesto de que estas cuentas se dedican tan solo a responder consultas y reclamos de los clientes, es decir, estas cuentas no publican mensajes de publicidad, concursos, avisos u opiniones en general, ya que para eso están las otras cuentas que poseen estas empresas en Twitter. Tomando esto en consideración se puede definir el número máximo de tweets (es decir, el peor caso) que le llegan a cada una de estas cuentas, esto implica suponer que cada tweet que publica una empresa en Twitter está respondiendo uno de un cliente, y que estas conversaciones tan solo tienen un mensaje de parte de los clientes. Por lo que finalmente bajo este supuesto la cantidad de tweets que llegan a estas empresas es a lo más la misma cantidad que ésta ha emitido.

Considerando entonces este número como el universo de tweets, se puede construir una muestra

representativa, que consta de 1.501 tweets, la cual tiene un error asociado del 3 % y un 98 % de confianza.

Además, como cada empresa posee una cantidad diferente de tweets, tanto el total de historia, como los que le llegan diariamente de parte de los usuarios, es preferible realizar un muestreo estratificado, que equivale a seleccionar una cantidad de tweets proporcional al total de cada empresa. El muestreo estratificado proporcional produce siempre menor o igual error muestral que el muestreo aleatorio simple, es decir, es más preciso.

Finalmente para el set de entrenamiento serán considerado un total de 2.000 tweets, debido a la metodología de selección de los datos que se utilizará, la cual será descrita en la siguiente sección. Dado esto, se utilizaron 1.000 tweets, provenientes del proceso de extracción (1) y (2) (Search keyword y La Gorda respectivamente), ya que representan el estado más actual de las empresas y es el conjunto de datos que más se acerca a la realidad. Además se consideraron otros 1.000 tweets, extraídos del proceso (3) (crawler se usuarios chilenos), esto con el fin de agregarle antigüedad al set, y que no presente sesgos o estacionalidad en el contexto de este último periodo. Esto da un total de 2.000 tweets que es superior a la cantidad definida para el set de datos. Esto se debe a que se eliminaran algunos tweets debido a la metodología de etiquetado a utilizar, la cual será explicada más adelante.

4.6. Definición de Categorías

Una vez terminada la extracción de los tweets, se analizó una muestra de ellos de manera exploratoria para estudiar su contenido y poder definir finalmente en que categorías se clasificarán. A modo general, se logró identificar en una primera etapa, que además de mensajes con carácter de reclamo, existen tweets con otras naturalezas: consultas, agradecimientos, entre otros. y en una segunda etapa, se descubrieron nuevas categorías adicionales a los cinco servicios de este rubro. A continuación se detalla de mejor forma lo explorado.

Como primer análisis de los datos se tuvo que existe una cantidad significativa (más del 10 %) de mensajes que representan consultas sobre los servicios y productos de las empresas. Estos mensajes presentan un carácter neutro y buscan una respuesta por parte de la empresa con respecto a algún producto o servicio. Por esto, se decidió definir una nueva categoría en relación a estos mensajes.

Se pudo observar que una parte muy pequeña representan tweets de *No Reclamo*, estos corresponden a mensajes de agradecimiento por parte de los clientes que han resueltos sus problemas o mensajes despectivos, los cuales no buscan una respuesta por parte de la empresa, tan solo tienen la finalidad de realizar un descargo y no merecen la pena crear una nueva categoría para diferenciarlos, ya que corresponden a un porcentaje muy pequeño.

Como tercera observación se identificó que existe una gran cantidad de tweets que hacen alusión a que la empresa responda mensajes anteriores emitidos por el autor, o que la empresa se contacte vía mensajes privados con este. En mayor detalle en estos mensajes los clientes solicitan que la empresa responda otros mensajes (que no se pueden identificar mediante el tweet en cuestión). Un Ejemplo de este tipo de mensaje se muestra a continuación:

@AyudaMovistarCL les envié un dm! Por favor léanlo[14]

En base a este nuevo contexto encontrado en los tweets, es que se decide definir un nuevo tópico en la detección inicial, el cual corresponde a los mensajes de: *Solicitar Respuesta a otros mensajes*.

Finalmente en la etapa de detección quedan definidos los siguientes tópicos o naturalezas de los tweets:

1. Reclamo
2. Consulta o solicitud de ayuda
3. Solicitud de respuesta a otros mensajes
4. No reclamo / Otro

En una segunda etapa se analizaron de manera exploratoria los tweets con respecto a las categorías en los cuales se enmarcan. Se buscó ver si estaban presentes las 5 categorías que definen la SUBTEL y SERNAC en relación a los servicios de las telecomunicaciones, y a su vez investigar si existen nuevas categorías.

Como primera observación se logró corroborar que las categorías correspondientes a los 5 servicios definidos están presenten en los tweets que se envían a las cuentas de ayuda y soporte. En general, la categoría con mayor cantidad de mensajes es *Internet Fija*, seguido por *Internet Móvil*, *Telefonía Móvil*, *Televisión* y *Telefonía Fija* respectivamente.

Además de las cinco categorías descritas anteriormente, se pudieron distinguir cuatro nuevas:

- *Pagos/Facturas*: corresponden a mensajes con relación a cobros, pagos, problemas con la factura de la cuenta, entre otros.
- *Visita técnica*: representan tweets con relación a problemas con la visitas técnicas, ya sea para instalar, reparar servicios, por la no aparición de los técnicos a la hora programada, etc.
- *Pagina Web / App móvil*: Son mensajes relacionados con problemas o consultas sobre la página web de la empresas, como fallas de sesión, disponibilidad del servicio, etc. Además de mensajes relacionados con la aplicación móvil para Smartphones de la empresa.
- *Ninguno de los anteriores*: Esta categoría fue introducida debido a que existen varios mensajes que no encajan en ninguna de las categorías nombradas anteriormente, y es necesario dejarlos a un lado y no forzarlos a entrar en otra. Principalmente corresponden a mensajes de reclamo en donde no es posible identificar el servicio al cual se refiere, o mensajes sobre mala atención de parte de la empresa.

En resumen, se añadieron 4 nuevas categorías para el análisis, quedando un total de 9, las cuales se enumeran a continuación:

1. Telefonía Móvil
2. Telefonía Fija

[14] dm corresponde a un mensaje directo / privado.

3. Internet Fija
4. Internet Móvil
5. Televisión
6. Facturas y Pagos
7. Visita Técnica
8. Pagina Web / App móvil
9. Ninguno de los anteriores

En conclusión se logró identificar nuevos tópicos en los cuales agrupar los tweets, tanto en su naturaleza como el servicio que hacen alusión. Finalmente el algoritmo a desarrollar en una primera etapa debe diferenciar entre 4 tópicos, y en una segunda etapa, los que fueron identificados como *Reclamos* deben ser categorizados dentro de alguna de las 9 clases que hacen referencia a los servicios de este rubro de las telecomunicaciones.

4.7. Etiquetado del set de datos

Para el proceso de etiquetado se utilizó una metodología similar a la realizada en [64-66] con el fin de obtener un etiquetado consistente, en donde se utiliza la sabiduría de las masas para perfeccionar el contenido del set de datos. Por lo que las opiniones recolectadas de Twitter se categorizan utilizando clasificadores humanos para determinar a que tópico pertenecen.

La principal ventaja que posee esta alternativa es que permite disminuir el sesgo presente en la interpretación de una sola persona acerca de un tweet, ya que al utilizar a múltiples evaluadores se están tomando puntos de vista diferentes, lo que genera un análisis más heterogéneo de la muestra.

4.7.1. Diseño del proceso de etiquetado

Los 2.000 tweets del set de entrenamiento se dividen en 10 grupos de 200 textos cada uno. Luego utilizando a 11 evaluadores humanos[15] se analizan los tweets de manera individual, de modo que cada uno es analizado y etiquetado por dos personas distintas. La distribución de los tweets se realizó de modo que una persona etiquetó la muestra completa, es decir, los 2.000 tweets y su interpretación se contrastó con los otros 10 etiquetadores. La figura 4.7 visualiza la distribución de los tweets para un mejor entendimiento.

Este proceso de etiquetado se basó en la implementación de una página de web de etiqueta-

[15] corresponden a personas de entre 22 y 30 años de edad que poseen y utilizan la red social de Twitter, por lo que son capaces de identificar todos los elementos que presentan los tweets con el objetivo de comprender de manera completa el mensaje.

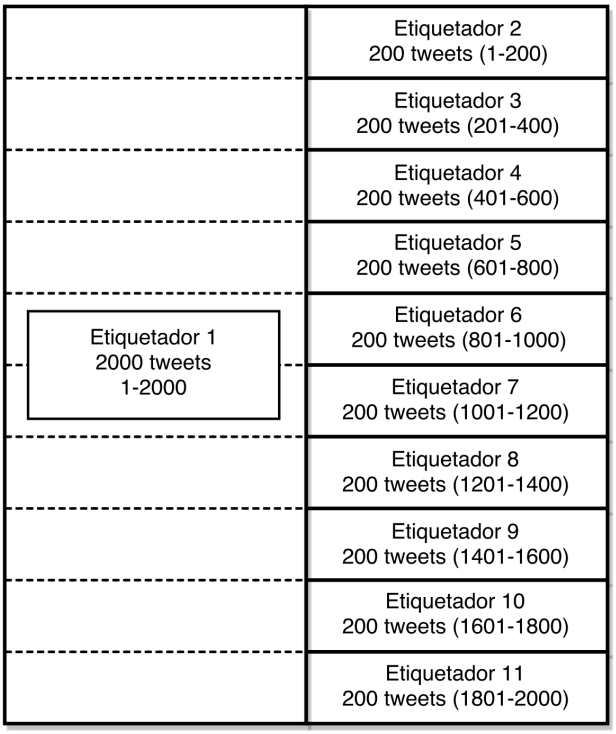

Figura 4.7: Distribución de tweets para etiquetado.
Fuente: Elaboración propia.

do en [64], con las modificaciones respectivas para este caso. La imagen 4.8 muestra la página implementada para etiquetar los tweets.

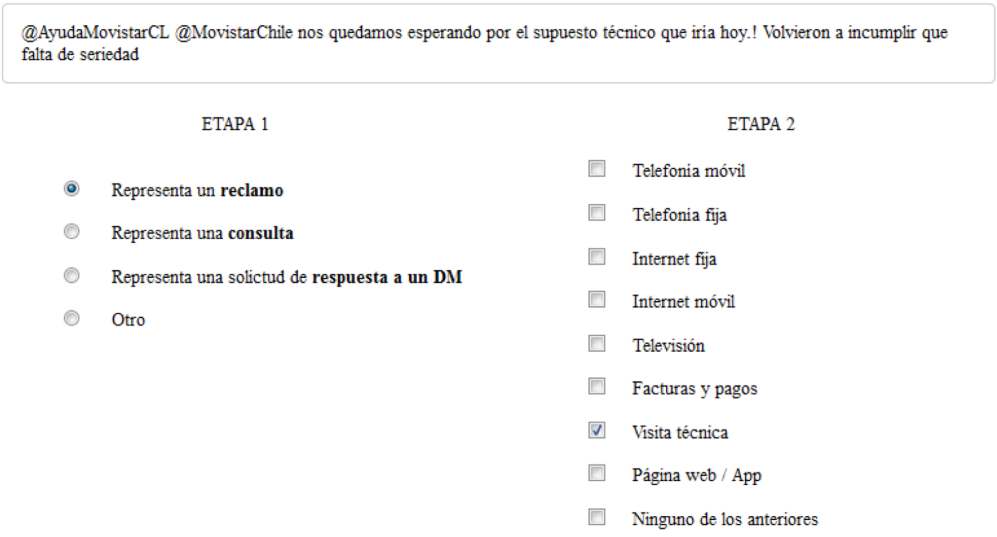

Figura 4.8: Página web de etiquetado de tweets implementada.
Fuente: Elaboración propia.

Se diseñó un conjunto de reglas para el proceso de etiquetado de tweets, con el objetivo de que todos los etiquetadores siguieran el mismo proceso para identificar a que clase pertenecía cada tweet. Además, se les presentaron ciertos ejemplos de casos ya etiquetados para que tuvieran un

mejor entendimiento de la tarea a realizar. Las reglas diseñadas son descritas a continuación:

1. Se le indica a los etiquetadores que deben utilizar su juicio como usuario de Twitter[16].

2. El usuario debe etiquetar el tweet en dos etapas: en un comienzo debe identificar si representa un reclamo, consulta o ayuda, respuesta a DM u otro. Y en una segunda parte debe categorizar el texto en base 8 categorías, y de no satisfacer ninguna debe seleccionar *Ninguna de las anteriores*.

3. En caso de ambigüedad en alguna opción en la segunda etapa del etiquetado, los usuarios la deben marcar sólo si es directamente inferible desde el tweet.

Para un etiquetado de estas características, en donde se utilizan a dos evaluadores para clasificar una muestra, existen dos principales alcances o alternativas a seguir para decidir finalmente a que clase pertenece la muestra en cuestión, si es que los evaluadores se encuentran en desacuerdo en sus decisiones.

1. La primera opción frente a una disyuntiva es dejar que un tercer evaluador dirima el desacuerdo, es decir, la elección de esta persona será la que finalmente decida a que categoría pertenece el tweet. Esta alternativa tiene varios inconvenientes, ya que implica que luego de analizar todos los datos (finalizada una primera etapa de etiquetado) se deben seleccionar aquellas muestras que posean desacuerdo, para diseñar y ejecutar una segunda iteración del proceso de etiquetado con estos tweets. Para este caso en particular en donde se tienen múltiples categorías, además de que cada tweet debe ser identificado en dos partes, podría ocurrir que este tercer evaluador escoja una categoría distinta a la de los dos primeros. Por lo que esta primera alternativa presenta dificultades en cuanto al tiempo necesario en realizar una segunda iteración y el encontrar a nuevos evaluadores dispuestos a realizar este trabajo.

2. Como segunda alternativa al enfrentarse a una muestra que haya sido clasificada de manera distinta por dos evaluadores, se puede decidir por desecharla, ya que es considerada como una muestra ambigua y por lo tanto el clasificador a construir podría aprender de mala forma si es que esta muestra es finalmente incorporada en el set de entrenamiento. Aunque dicha muestra no sea utilizada para entrenar el modelo, debe ser evaluada posteriormente ya que representa a un ejemplo de la realidad, por lo que no puede dejarse de lado y no ser considerada en los resultados del modelo.

Considerando el tiempo requerido para el proceso de etiquetado y la calidad de los datos que finalmente alimentarán el modelo de clasificación, es que se decide por utilizar el segundo alcance en relación al que hacer con las muestras que posean conflicto. Habiendo tomado esta elección, es necesario destacar qué dado que el proceso de etiquetado posee dos etapas, puede existir discordia entre los evaluadores en una o las dos etapas, por lo que se se formarán dos set de entrenamiento, los cuales pueden no contener exactamente los mismos tweets. Por ejemplo, dos evaluadores están de acuerdo en que una muestra representa un *Reclamo*, sin embargo, difieren en la categoría, uno dice que es sobre *Telefonía móvil* y el otro *Internet móvil*, por lo que este tweet será contemplado en el set de entrenamiento para la detección de reclamos, pero no así para la categorización.

[16]conocimiento de los elementos presentes en los mensajes de Twitter

4.7.2. Resultados del proceso de etiquetado

Como resultado del proceso de etiquetado, se obtuvieron 1.834 tweets analizados por los etiquetadores. Estos datos fueron filtrados para formar finalmente el set de entrenamiento del clasificador. Este proceso, como se mencionó en el apartado anterior, siguió la metodología más rápida para escoger las muestras válidas. Recordando, este procedimiento consiste en descartar las muestras que presenten desacuerdo entre los dos etiquetadores. Para entrenar el modelo se pretende utilizar solo los datos que tengan un completo acuerdo entre los etiquetadores, con el objetivo que el modelo aprenda a diferenciar las clases de la mejor forma, por lo que los datos que posean desacuerdo serán descartados para entrenar el modelo, sin embargo estos serán analizados posteriormente para evaluar el comportamiento del modelo desarrollado en estos datos ambiguos y así observar como se desempeñará el clasificador en los nuevos datos que se presenten en el futuro.

Además, como se estipuló en el "diseño del proceso de etiquetado", las opciones de la etapa 1, de ahora en adelante llamada *Etapa de Detección* (identificar la naturaleza del tweet) son excluyentes, mientras que en la etapa 2, de ahora en adelante llamada *Etapa de Categorización* (identificar la categoría del tweet) las opciones no son excluyentes. Por esto se desarrolló un set individual para cada una, y en consecuencia cada clasificador tendrá un set de entrenamiento distinto, tanto en contenido como en su tamaño.

En lo que sigue se describirá el procedimiento llevado a cabo para la selección de muestras para cada uno de los set de entrenamiento.

4.7.2.1. Set de entrenamiento Etapa de Detección

De los 1.834 tweet etiquetados para esta etapa, 266 de ellos presentaron conflicto entre lo dicho por ambos etiquetadores, por lo que en base a la metodología escogida, estos fueron removidos, quedando un total de 1.568 tweets en donde ambos etiquetadores coincidieron en su elección.

Por otra parte se tienen las métricas de acuerdo entre los evaluadores. En la tabla 4.6 se detalla el indicador para la primera etapa de etiquetado, la cual corresponde a la detección de reclamo, consulta, solicitud de respuesta y otro. Cabe destacar que tan solo se calcula una métrica *Kappa de Fleiss* en este caso, ya que las clases son excluyentes, es decir, el tweet puede estar asociado a tan solo una de estas cuatro opciones. La interpretación para el valor asociado al acuerdo de esta etapa es "Razonable a Bueno", esto indica que la tarea de distinguir una opinión de reclamo no es una tarea tan sencilla, inclusive para un humano.

Acuerdo Relativo	Kappa de Fleiss
0.85	0.76

Tabla 4.6: Medidas de Acuerdo Etiquetado Etapa de Detección.
Fuente: Elaboración propia.

4.7.2.2. Set de entrenamiento Etapa de Categorización

Analizando ahora los datos obtenidos de la segunda etapa del proceso de etiquetado, también se tienen 1.834 tweets, para este caso los datos en desacuerdo son 262, quedando 1.572 en donde ambos etiquetadores coincidieron en su elección. Dentro de los tweets etiquetados, 224 de estos fueron etiquetados en más de una categoría, sin embargo los que presentan acuerdo por parte de ambos etiquetadores son tan solo 96. Lo que indica que identificar múltiples clases para una muestra es una tarea bastante compleja para un humano, ya que existe tan solo un 42 % de acuerdo.

Tal como se visualiza en la tabla 4.7, todas las categorías muestran un coeficiente *Kappa de Fleiss* que fluctúa entre 0,7 y 0,9, a excepción de "Página Web/App" y "Internet Móvil". Esto se debe a que a pesar de tener los más altos acuerdos relativos, los datos no poseen mayor variabilidad, por lo que el coeficiente es castigado directamente. En general, a excepción de estas dos clases, la fuerza de acuerdo es "Excelente", lo que indica que los datos son fácilmente distinguibles por humanos. Para este caso se identifica una métrica para cada categoría, ya que estas no son excluyentes.

Categoría	Acuerdo Relativo	Kappa de Fleiss
Telefonía Móvil	0.95	0.81
Telefonía Fija	0.97	0.79
Internet Fija	0.96	0.91
Internet Móvil	0.96	0.76
Televisión	0.97	0.92
Cuenta / Factura	0.97	0.77
Visita Técnica	0.98	0.86
Página Web / App	0.98	0.71
Ninguno	0.93	0.85

Tabla 4.7: Medidas de Acuerdo Etiquetado Etapa de Categorización.
Fuente: Elaboración propia.

Para este desafío de clasificar una muestra en varias clases, donde potencialmente podría ser más de una simultáneamente, se dispone a construir un algoritmo que posea 8 clasificadores binarios, también conocidos como *One-SimpleClass*. De este modo se desarrollará y evaluará un clasificador para cada clase, a excepción de la categoría "Ninguno de los Anteriores", debido a que si una muestra no pertenece a ninguna de las 8 clases, se categoriza como "Ninguna".

4.7.3. Resumen

Finalmente en base a lo expuesto anteriormente es que se justifica la elección de una cantidad superior a la definida como una muestra representativa del universo de tweets de las cuatro empresas a considerar,ya que existía una alta probabilidad que hubiesen desacuerdos en las elecciones de los evaluadores, por lo que una parte de los 2.000 tweets serían descartados, dando así una cantidad menor y más cercana al tamaño de muestra calculado en un comienzo para entrenar los modelos.

En segundo lugar los coeficientes de *Kappa de Fleiss* calculados muestran que existe un acuerdo "Bueno a Excelente" entre los etiquetadores, que indica el desafío de identificar las opiniones para humanos no es complejo. Ahora es necesario evaluar como actúa un computador frente a esta problemática.

En resumen, del proceso de etiquetado se obtuvieron 1.568 datos para el set de entrenamiento en la etapa de detección. La tabla 4.8 muestra en detalle la cantidad de datos para cada clase.

Tópico	Cantidad
Reclamo	809
Consulta / Ayuda	518
Respuesta (DM)	181
Otro	60

Tabla 4.8: Cantidad de datos por clase - Etapa de Detección.
Fuente: Elaboración propia.

Por otro lado, para la etapa de categorización se obtuvo un set de 1.572 datos. Las tabla 4.9 muestran la cantidad de datos para cada clase

Tópico	Cantidad
Telefonía Móvil	203
Telefonía Fija	79
Internet Fija	430
Internet Móvil	103
Televisión	198
Cuenta / Factura	90
Visita Técnica	70
Página Web / App	24
Ninguno	494

Tabla 4.9: Cantidad de datos por clase - Etapa de Categorización
Fuente: Elaboración propia.

Capítulo 5

Modelo de Detección y Clasificación de Reclamos

Este capítulo tiene como objetivo desarrollar y evaluar diferentes enfoques y modelos de clasificación supervisada para resolver el problema de detectar y categorizar opiniones de reclamos presentes en Twitter, y finalmente escoger el modelo que mejor desempeño entregue para integrarlo en la plataforma web de OpinionZoom.

Recapitulando lo abordado por las investigaciones realizadas en este ámbito de clasificación de opiniones de Twitter del capítulo III, se pudo observar que el enfoque mas utilizado es Bag-Of-Words, el cual permite representar un set de documentos como una matriz de frecuencias u ocurrencias de tokens. En este tipo de técnica el mayor problema es que el set de potenciales términos utilizados es enorme, mas aún en un dominio como Twitter donde existen múltiples variaciones lingüísticas y elementos en el texto, tales como abreviaciones de palabras, palabras mal escritas, jergas, entre otros. Por lo tanto este set debe ser reducido para construir un clasificador práctico y en consecuencia el preprocesamiento del texto y la selección de los términos más importantes a ser utilizados constituyen una etapa fundamental [67].

Como primer paso en la problemática a resolver se tiene el identificar si una opinión presenta carácter de reclamo de manera automática, y en segundo lugar se tiene el clasificar estas opiniones de reclamo en clases predefinidas, que para este caso representan servicios en el rubro de las telecomunicaciones. Para abordar este desafío se utilizaron técnicas de Data Mining, más en específico Machine Learning, que permiten a las computadoras aprender de los datos para predecir nuevos no antes vistos. Pero antes que todo, es necesario que el texto sea representado en una estructura definida tal que los algoritmos de Machine Learning puedan interpretar los datos, ya que no están diseñados para tratar con texto de una forma directa y por ende se deben realizar transformaciones a una estructura entendible para que estos algoritmos puedan operar.

En lo que sigue se evaluarán estos dos enfoques de representación de texto, aplicados al contexto de detección y categorización de opiniones de reclamos en Twitter para el rubro de las telecomunicaciones, en los set de datos construidos y explicados en el capítulo anterior.

5.1. Especificaciones de la Implementación

Es importante identificar los recursos utilizados en la implementación de los modelos a evaluar, debido a que los algoritmos de Machine Learning son pesados computacionalmente y esto condiciona la factibilidad de los que se puedan realizar. Además las especificaciones permiten realizar una mejor replicación de los resultados que se obtengan.

La implementación de los modelos de Machine Learning se llevó a cabo mediante la librería WEKA [56] en su versión número 3, la cual está construida en el lenguaje de programación Java. Considerando que el núcleo de la plataforma OpinionZoom está desarrollado en este mismo lenguaje, se decide por realizar todo (preprocesamiento de texto y construcción del clasificador) en este lenguaje.

1. **Hardware:** Para todo el desarrollo de este trabajo se utilizó un notebook con las siguiente características:

   - MacBook Pro (Retina, Mid 2012)
   - Procesador Intel Core i7 2.7 GHz
   - Memoria RAM 16 GB 1600 MHz DDR3
   - Gráficos NVIDIA GeForce GT 650M 1024 MB, Intel HD Graphics 4000 1536 MB

2. **Software:** El software en el cual se diseñaron los algoritmos y posteriormente el clasificador fue:

   - Sistema Operativo: macOS Sierra 10.12.3
   - Ambiente de desarrollo: NetBeans IDE 8.2
   - Java(TM) SE 1.8.0_101-b13

5.2. Detección de Reclamos

El primer problema a resolver es el identificar las opiniones de reclamos presentes en Twitter. Tal como se analizó en el capítulo anterior, se pudo observar que además de opiniones con naturaleza de *Reclamo* y *No Reclamo* existen otras dos: *Consultas* y *Solicitudes de respuestas a otros mensajes (DM)*. Por lo que ahora el problema se transforma en identificar a cual de estas cuatro clases pertenece el tweet.

Tan solo para esta Etapa de Detección se decidió evaluar el comportamiento de la *Polaridad* de los tweets como una variable adicional para clasificar las opiniones de reclamo. Esto bajo el supuesto de que una opinión de reclamo posee un carácter negativo, ya que se define como una disconformidad de parte del autor del mensaje. Esta variable de polaridad fue obtenida a través de la *PAPI*[1] desarrollada en [68], la cual es utilizada por la plataforma de OpinionZoom, por lo que no considera un problema mayor en caso de ser necesaria.

[1] Polarity API

5.2.1. Modelo Bag-Of-Words

Como paso inicial al tratar con texto se deben realizar las técnicas de preprocesamiento vistas en el capítulo II. Estas son *Tokenización*, *Borrado de Stopwords*, *Stemming* y *N-grams*. Además de estas técnicas generales, se realizaron otras técnicas específicas para trabajar con texto proveniente de Twitter, y que son previas a las 4 mencionadas antes. La figura 5.1 muestra el flujo de todas las operaciones que se le realizan al texto durante la etapa de preprocesamiento. Es importante mencionar que en los modelos a evaluar el flujo de operaciones puede o no considerar todas estas técnicas, pero será explicado con mayor detalle más adelante.

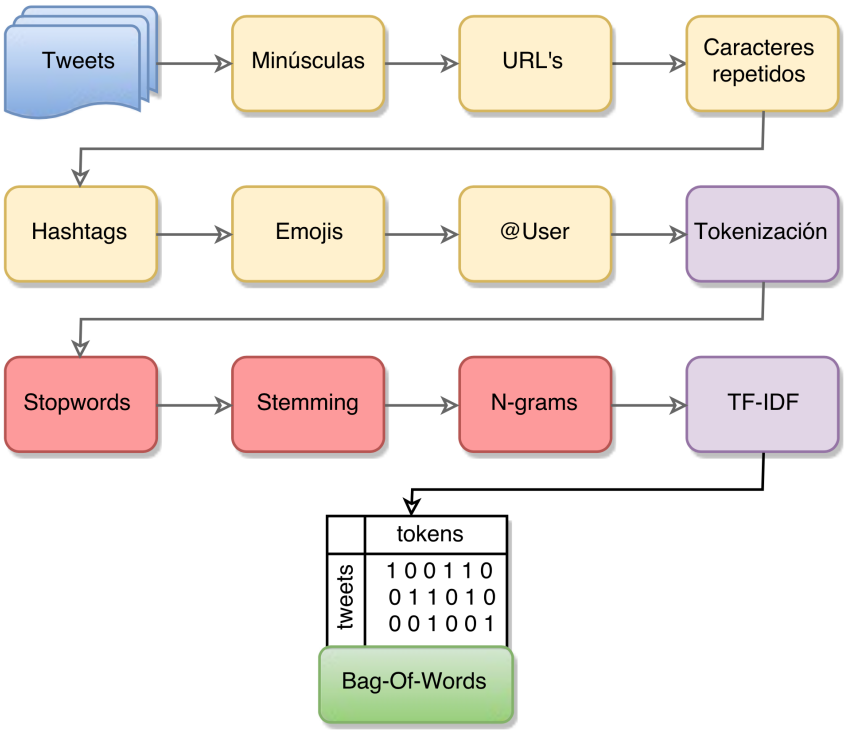

Figura 5.1: Preprocesamiento de texto
Elaboración propia.

A continuación se define cada uno de estos pasos:

1. **Minúsculas:** Todo el texto es transformado a minúsculas para estandarizar los caracteres.

2. **URL's:** Si el texto contiene un link, este es transformado al token "URL". Por ejemplo el siguiente tweet:

   @AyudaMovistarCL favor su ayuda.. https://t.co/rs1wLpjdQN

   es transformado en:

   @AyudaMovistarCL favor su ayuda.. URL

3. **Caracteres repetidos:** Se eliminan del tweet caracteres repetidos en las palabras, como por ejemplo: "Holaaaaaaa" es transformado en "Hola".

4. **Hashtags:** A estas palabras se les quita el simbolo # que precede al termino, por ejemplo el hashtag “#internet” queda tan solo como “internet”.

5. **Emojis:** Los emoticones presentes en el texto son transformados a la palabra que representan. Por ejemplo el emoticon 😡 es transformado al token “rage”.

6. **@User:** Las menciones a los usuarios de las cuatro cuentas analizadas son eliminados del texto. Estos son: @AyudaMovistarCL, @entel_ayuda, @VTRSoporte y @miclaro_cl.

7. **Tokenización:** El texto fue tokenizado utilizando los siguientes separadores además de los espacios en blanco:

$$\begin{pmatrix} ; & , & . & ? \\ ¿ & ¡ & ! & \backslash \\ " & ' & ‘ & ´ \\ / & : & - & + \\ * & \% & \# & (\\ ) & \$ & \& & \{ \\ \} & [&] & < \\ > & \sim & | & @ \end{pmatrix}$$

8. **Stopwords:** Para la remoción de Stopwords fue utilizada la lista que provee Snowball, la cual cuenta con 325 palabras para el idioma español. En **??** se muestra la lista completa de Stopwords utilizada.

9. **Stemming:** Para realizar el proceso de Stemming fue utilizada la libreria que posee Snowball en su idioma español.

10. **N-grams:** Son utilizados Uni-grams y Bi-grams solamente, ya que una cantidad mayor genera peores resultados.

11. **TF-IDF:** Se calculó el peso mediante la ponderación TF-IDF para cada término en cada uno de los tweets.

12. **Bag-Of-Words:** Finalmente se tiene el modelo de Bag-Of-Words el cual tiene la estructura necesaria para aplicar los algoritmos de Machine Learning de una forma directa.

Teniendo entonces listo el modelo de Bag-Of-Words, se pueden aplicar los algoritmos de Machine Learning, pero no cualquiera, debe ser capaz de clasificar en más de una clase, por lo que no todos los algoritmos son aptos para realizar esta tarea. La elección de estos algoritmos tuvo como referencia los algoritmos que se utilizan en las investigaciones vistas en el capitulo III, en conjunto con los limites de procesamiento del hardware a utilizar. Por ejemplo se descartó evaluar los algoritmos de *Redes Neuronales* y *Regresión Logística* dada la alta cantidad de variables del clasificador a construir. Finalmente, para esta Etapa de Detección fueron analizados los siguientes cuatro algoritmos:

1. Multinomial Naive Bayes

2. Multi-Class Support Vector Machines

3. Random Forest

4. Decision Tree

Para cada uno de estos algoritmos fueron evaluadas distintas combinaciones de preprocesamiento de texto sumado a la incorporación de la variable Polaridad del tweet, en mayor detalle se variaron los tres pasos en rojo de la figura 5.1 y se evaluó la inclusión de la variable *Polaridad*. La tabla 5.1 muestra las 16 combinaciones realizadas para cada uno de los cuatro algoritmos.

Configuración	Remoción de Stopwords	Stemming	Polaridad	Bi-grams
1	✓	✓	✓	✓
2	✓	✓	✓	-
3	✓	✓	-	✓
4	✓	✓	-	-
5	✓	-	✓	✓
6	✓	-	✓	-
7	✓	-	-	✓
8	✓	-	-	-
9	-	✓	✓	✓
10	-	✓	✓	-
11	-	✓	-	✓
12	-	✓	-	-
13	-	-	✓	✓
14	-	-	✓	-
15	-	-	-	✓
16	-	-	-	-

Tabla 5.1: Configuraciones de preprocesamiento de texto - Etapa de Detección.
Fuente: Elaboración propia.

Es importante destacar que no se realizó optimización de los parámetros de los modelos, debido a que no es una única configuración de preprocesamiento a evaluar, y de realizar optimización particular para cada uno podría producirse un sesgo en los resultados, ya que las diferencias en el desempeño de cada uno no son significativas. Más importante, de realizar optimización de parámetros, los resultados no serían comparables, ya que el algoritmo aplicado podría no ser el mismo. En conclusión, se utilizaron los parámetros por defecto de los clasificadores para todas las combinaciones evaluadas.

En lo que sigue se exponen los resultados para cada uno de los 4 algoritmos de Machine Learning evaluados para la Etapa de Detección. Cada configuración fue evaluada utilizando 5-Fold cross validation, esto es entrenar el modelo con el 80 % de los datos y testearlo con el 20 % restante. La decisión de particionar el set de datos de esta forma se basó en las distintas investigaciones analizadas en el capítulo III, donde la mayoría segmentaba los datos en estas proporciones.

Como regla para decidir que configuración es la que mejor desempeño presenta se considera el

valor de los indicadores en el siguiente orden:

1. F-Measure (F1)
2. Accuracy
3. Precision y Recall de la clase *Reclamo*
4. Precision y Recall de las otras clases

Donde no está demás aclarar que mientras más grande el valor, mejor es el indicador.

5.2.1.1. Multinomial Naive Bayes

Como primer algoritmo de Machine Learning se analizó Naive Bayes en su versión para multiclases. Este no entrega opciones para variar sus parámetros, puesto que es un modelo que se basa en probabilidades condicionales, las cuales no tienen parámetros. Se puede realizar un ajuste mediante la "Corrección de Laplace", la cual se encarga de los casos en que se generan probabilidades cero para ciertas variables y estas tienen un alto nivel de influencia en el modelo, sin embargo, la librería de WEKA no posee esta capacidad, por lo que no se realiza esta corrección.

La tabla 5.2 visualiza los resultados obtenidos para cada una de las 16 combinaciones de preprocesamiento al aplicar el algoritmo de Naive Bayes Multinomial de la librería WEKA. Esta tabla muestra la *Precision* y *Recall* para cada clase, el nivel de *Accuracy* general del clasificador y el indicador *F-Measure* (F1). Los valores Subrayados representan los 3 valores más altos para cada indicador con el fin de identificar las mejores configuraciones de preprocesamiento de texto.

De los resultados de este modelo se puede concluir que conservar las Stopwords mejora en alrededor de un 2% el nivel Accuracy de los modelos, lo que indica que estas palabras si aportan valor para determinar si una opinión es considerada un reclamo. También se puede observar que la utilización de Uni-grams se desempeña de mejor forma que considerar la unión de términos consecutivos con el fin de incorporar mayor contexto a las variables. Esto quiere decir que este para este algoritmo de Naive Bayes tan solo interesa la aparición de las palabras de manera independiente en el texto para discriminar entre las clases.

De todas las configuraciones de preprocesamiento de texto analizadas para el algoritmo de Naive Bayes, la que mejor desempeño obtuvo fue la número 10, donde no fueron removidas las Stopwords, se realizó Stemming, se incorporó la Polaridad y se utilizaron Uni-grams. Este modelo presenta el nivel de *Precision* más alto para la clase de mayor importancia, "Reclamo", además alcanza el nivel de *Acurracy* (73.0%) y *F-Measure* (0.731) más alto de todas las configuraciones evaluadas.

Con el fin de apreciar de mejor forma los resultados obtenidos en esta configuración, la tabla 5.3 muestra la matriz de confusión para este escenario.

De la matriz de confusión se puede extraer que el clasificador es más asertivo para la clase *Reclamo*, seguido de *Consulta*, *Respuesta* y *Otro* respectivamente. Esto se debe a la disparidad en

Config.	Clases (%)								Accuracy (%)	F1
	Reclamo		Otro		Consulta		Respuesta			
	P	R	P	R	P	R	P	R		
1	78.2	71.7	27.6	45.0	69.9	69.1	65.3	77.9	70.5	0.711
2	78.4	74.0	30.7	38.3	70.2	69.7	66.0	78.5	71.7	0.720
3	78.1	71.4	28.3	46.7	70.0	68.3	71.7	70.2	70.2	0.707
4	77.8	74.5	29.6	35.0	70.6	68.5	64.8	78.5	71.5	0.717
5	77.4	71.9	28.2	40.0	67.4	69.5	69.0	75.1	70.3	0.707
6	77.1	73.3	30.2	31.7	66.8	70.1	72.1	75.7	71.1	0.712
7	77.8	71.3	26.4	40.0	67.8	69.5	67.2	75.7	70.0	0.705
8	77.3	72.9	28.8	31.7	66.8	70.8	71.6	75.1	70.9	0.711
9	78.5	78.9	38.6	56.7	71.3	64.9	68.9	74.6	72.9	0.731
10	79.4	78.0	36.8	46.7	71.6	67.2	64.9	75.7	73.0	0.731
11	78.5	78.1	35.5	55.0	71.2	65.1	68.5	74.6	72.5	0.728
12	79.0	78.0	36.8	46.7	71.8	66.8	64.9	75.7	72.8	0.730
13	79.2	77.3	37.9	55.0	69.3	66.8	70.5	75.1	72.7	0.730
14	79.5	76.1	40.5	50.0	68.6	67.8	66.2	75.7	72.3	0.725
15	78.8	76.5	38.5	58.3	68.7	66.4	71.1	74.6	72.3	0.725
16	78.9	75.4	38.7	48.3	68.3	67.8	66.0	75.1	71.8	0.720

Tabla 5.2: Resultados Multinomial Naive Bayes - Etapa Detección.
Fuente: Elaboración propia.

la cantidad de datos de cada clase. Al existir menos muestras en una categoría, el clasificador no alcanza a considerar toda la variabilidad de los datos, ya que existen términos con poca frecuencia y por ende el clasificador no sabe a que clase pertenecen.

5.2.1.2. Multi-Class Support Vector Machines

El segundo algoritmo de Machine Learning analizado es SVM. Como se mencionó anteriormente, se utilizaron los parámetros por defectos de la librería WEKA, para el caso de SVM estos se detallan en la tabla 5.4. Al igual que en el caso anterior se evaluó el desempeño de este clasificador en las 16 configuraciones de preprocesamiento de texto. La tabla 5.5 muestra los resultados.

		Clase predicha				
		Reclamo	Otro	Consulta	Respuesta	Recall
clase real	Reclamo	631	34	122	22	78.0%
	Otro	19	28	2	11	46.7%
	Consulta	120	9	348	41	67.2%
	Respuesta	25	5	14	137	73.0%
	Precision	79.4%	36.8%	71.6%	64.9%	

Tabla 5.3: Matriz de Confusión - Etapa de Detección - Naive Bayes - Configuración Nª10.
Fuente: Elaboración propia.

Parámetro	Valor
Complexity (C)	1
Tolerancia (L)	10^{-3}
Error (ξ)	10^{-12}
Semilla	1
Kernel	PolyKernel

Tabla 5.4: Parámetros Support Vector Machines.
Fuente: Elaboración propia.

Config.	Clases (%)								Accuracy (%)	F1
	Reclamo		Otro		Consulta		Respuesta			
	P	R	P	R	P	R	P	R		
1	77.9	82.8	42.1	26.7	75.9	71.6	77.9	77.9	76.4	0.760
2	76.6	80.5	44.1	25.0	70.8	69.1	80.9	79.6	74.5	0.741
3	77.9	82.8	41.0	26.7	76.0	71.6	77.9	77.9	76.4	0.760
4	76.6	80.7	43.8	23.3	71.1	69.3	80.9	79.6	74.6	0.742
5	76.9	81.7	40.0	30.0	72.8	69.7	79.8	74.0	74.9	0.746
6	75.9	81.1	48.6	28.3	69.9	67.4	80.0	75.1	73.9	0.734
7	76.9	81.7	38.6	28.3	72.8	69.9	79.8	74.0	74.5	0.746
8	75.6	80.8	45.9	28.3	70.0	67.0	80.0	75.1	73.6	0.732
9	80.7	83.6	38.7	48.3	76.1	72.6	85.1	75.7	77.7	0.778
10	78.2	83.2	57.4	45.0	73.5	69.7	81.1	75.7	76.4	0.762
11	80.9	83.8	39.2	48.3	76.4	73.0	85.1	75.7	77.9	0.780
12	78.2	83.6	56.5	43.3	73.9	69.5	80.1	75.7	76.5	0.762
13	80.1	83.7	41.6	53.3	75.0	71.2	86.4	73.5	77.2	0.773
14	77.7	82.4	49.2	48.3	71.7	67.6	80.4	72.4	75.1	0.749
15	80.1	83.6	41.6	53.3	74.8	71.2	86.4	73.5	77.2	0.773
16	77.7	82.3	47.5	48.3	71.7	67.6	80.9	72.4	75.0	0.749

Tabla 5.5: Resultados Multi-Class Support Vector Machines - Etapa Detección.
Fuente: Elaboración propia.

Del mismo modo que para el algoritmo de Naive Bayes, SVM muestra un aumento en todos los indicadores al mantener las Stopwords, también se puede observar que Bi-grams se desempeña mejor que Uni-grams. Por el lado de la variable *Polaridad* no se puede concluir nada, los resultados en muchos casos son los mismos. Finalmente, realizar Stemming mejora el rendimiento general de los modelos en cerca de 1 %.

De todas las configuraciones de preprocesamiento de texto analizadas para el algoritmo de Support Vector Machines, la que mejor desempeño obtuvo fue la número 11, donde no fueron removidas las Stopwords, se realizó Stemming, no se incorporó la Polaridad y se utilizaron Bi-grams. Este modelo presenta el nivel de *Precision* y *Recall* más alto para la clase de mayor importancia, "Reclamo", además alcanza el nivel de *Acurracy* (77.9 %) y *F-Measure* (0.780) más alto de todas las configuraciones evaluadas.

Con el fin de apreciar de mejor forma los resultados obtenidos en este escenario, se muestra la matriz de confusión en la tabla 5.6.

		Clas predicha				
		Reclamo	Otro	Consulta	Respuesta	Recall
clase real	Reclamo	678	25	96	10	83.8%
	Otro	26	29	4	1	48.3%
	Consulta	114	13	378	13	73.0%
	Respuesta	20	7	17	137	77.9%
	Precision	80.9%	39.2%	76.4%	85.1%	

Tabla 5.6: Matriz de Confusión - Etapa de Detección SVM - Configuración Nª11.
Fuente: Elaboración propia.

De la matriz de confusión se puede extraer que el clasificador es más asertivo para la clase *Reclamo*, seguido de *Respuesta*, *Consulta* y *Otro* respectivamente. Lo que se debe a la cantidad de datos en cada clase. Mientras más muestras posea, el clasificador incorpora mayor variabilidad de los datos. Además se puede apreciar que el clasificador se equivoca en las predicciones de forma pareja, es decir, proporcional a la cantidad de datos en cada una de las otras clases.

5.2.1.3. Random Forest

Como tercer algoritmo a evaluar se tiene a Random Forest. Se utilizaron los parámetros por defectos de la librería WEKA, para el caso de Random Forest estos se detallan en la tabla 5.7. Al igual que en el caso anterior se evaluó el desempeño de este clasificador en las 16 configuraciones de preprocesamiento de texto. La tabla 5.8 visualiza los resultados.

Parámetro	Valor
Iteraciones (I)	100
Número de árboles	1 (sin paralelismo)
Atributos (K)	0 (automático)
Min. de instancias	1
Min. de varianza	10^{-3}
Semilla	1
Profundidad	0 (sin límite)

Tabla 5.7: Parámetros Random Forest.
Fuente: Elaboración propia.

Al contrario de los dos algoritmos antes vistos, Random Forest no muestra una variación de rendimiento al incorporar o eliminar las Stopwords, el desempeño se mantiene similar en ambos escenarios. Por el lado de los N-grams, se puede apreciar claramente que utilizar Bi-grams empeora los resultados en alrededor de 2%. No se logra ver claramente si la variable *Polaridad* tiene un efecto positivo o negativo, en algunos casos mejora los indicadores, pero en otros lo empeora, por lo que no se puede concluir con respecto a la inclusión de esta variable. Finalmente, realizar Stemming mejora la *Precision* para la clase *Otro*.

Config.	Clases (%)								Accuracy (%)	F1
	Reclamo		Otro		Consulta		Respuesta			
	P	R	P	R	P	R	P	R		
1	74.0	88.9	37.3	31.7	79.3	60.6	88.0	72.9	75.5	0.749
2	75.8	89.2	67.9	31.7	78.9	66.2	86.8	72.8	77.6	0.769
3	74.1	87.4	32.9	40.0	78.4	58.9	88.2	74.0	74.6	0.743
4	76.1	88.4	46.2	30.0	78.5	66.4	88.2	74.0	77.2	0.767
5	73.0	88.0	32.4	38.3	77.6	57.5	91.2	69.1	76.9	0.734
6	76.2	87.6	47.5	31.7	77.7	67.4	90.5	74.0	77.2	0.767
7	74.7	87.0	23.8	40.0	80.0	59.5	90.6	69.6	74.1	0.742
8	75.5	86.8	40.7	36.7	77.8	65.1	86.8	72.4	76.0	0.757
9	72.2	94.2	52.3	38.3	82.1	53.3	91.7	66.9	75.4	0.741
10	73.9	93.3	70.0	35.0	81.0	60.2	91.7	66.9	77.1	0.761
11	73.0	93.8	51.1	38.3	82.3	54.8	92.0	70.2	76.1	0.750
12	73.8	93.6	70.0	35.0	81.2	59.3	92.5	68.5	77.1	0.761
13	71.4	94.6	54.3	41.7	82.9	50.4	91.1	68.0	74.9	0.735
14	73.7	94.7	73.3	36.7	84.4	59.5	93.2	68.5	77.8	0.768
15	72.8	93.1	48.0	40.0	82.4	55.2	90.5	68.5	75.7	0.747
16	73.4	93.9	65.5	31.7	83.1	58.9	91.2	68.5	77.0	0.759

Tabla 5.8: Resultados Random Forest - Etapa Detección.
Fuente: Elaboración propia.

De todas las configuraciones de preprocesamiento de texto analizadas para el algoritmo de Random Forest, la que mejor desempeño obtuvo fue la número 14, donde no fueron removidas las Stopwords, no se realizó Stemming, se incorporó la Polaridad y se utilizaron Uni-grams. Esta configuración presenta el nivel de *Acurracy* (77.8) más alto, y en *F-Measure* (0.768) se encuentra en segundo lugar, sin embargo para la clase "Reclamo" tiene el *Recall* (94.7) y *F1* (0.829) más altos.

Con el fin de apreciar de mejor forma los resultados obtenidos en esta configuración, la tabla 5.9 muestra la matriz de confusión para este escenario.

		Clase predicha				
		Reclamo	Otro	Consulta	Respuesta	Recall
clase real	Reclamo	766	2	39	2	94.7%
	Otro	35	22	1	2	36.7%
	Consulta	202	3	308	5	59.5%
	Respuesta	37	3	17	124	68.5%
	Precision	73.7%	73.3%	84.4%	93.2%	

Tabla 5.9: Matriz de Confusión - Etapa de Detección Random Forest - Configuración Nª14.
Fuente: Elaboración propia.

De la matriz de confusión se puede extraer que el clasificador es más asertivo para la clase *Respuesta*, seguido de *Consulta*, *Reclamo* y *Otro* respectivamente. Aunque considerando la medida F-Measure para cada clase, la que mejor se encuentra es la clase *Reclamo*, la más importante del problema. Además, se puede observar que este modelo presenta una *Precision* por sobre el 70% en

las cuatro clases, algo no visto en los dos algoritmos previamente analizados, donde la clase *Otro* siempre rondaba el 40 %.

5.2.1.4. Decision Tree

Como último algoritmo a evaluar se tiene a Decision Tree. Como se mencionó anteriormente, se utilizaron los parámetros por defectos de la librería WEKA, para el caso de Decision Tree estos se detallan en la tabla 5.10. Al igual que en los casos anteriores se evaluó el desempeño de este clasificador en las 16 configuraciones de preprocesamiento de texto. La tabla 5.11 visualiza los resultados.

Parámetro	Valor
Pruning Confidence (C)	0.25
Min. de instancias	2
Folds (N)	3
Semilla	1

Tabla 5.10: Parámetros Decision Tree.
Fuente: Elaboración propia.

Config.	Clases (%)								Accuracy (%)	F1
	Reclamo		Otro		Consulta		Respuesta			
	P	R	P	R	P	R	P	R		
1	76.2	81.7	40.0	6.7	69.2	72.6	85.0	69.1	74.4	0.733
2	76.2	80.2	50.0	13.3	68.4	72.0	80.0	68.5	73.6	0.728
3	76.3	82.1	40.0	6.7	69.5	72.6	85.7	69.6	74.6	0.736
4	76.3	80.0	33.3	10.0	68.7	72.4	79.5	68.5	73.5	0.727
5	75.1	80.2	37.5	15.0	69.5	71.2	81.9	67.4	73.3	0.726
6	75.4	80.1	36.4	13.3	70.5	71.4	76.5	68.5	73.3	0.726
7	75.9	80.3	33.3	15.0	70.2	73.2	84.1	67.4	74.0	0.734
8	74.5	80.0	33.3	15.0	69.5	68.9	78.5	68.5	72.5	0.719
9	77.2	80.6	41.7	16.7	70.3	74.3	84.1	70.2	74.9	0.743
10	75.8	80.3	41.7	16.7	69.3	71.8	83.3	69.1	73.8	0.732
11	76.9	81.1	42.9	15.0	70.3	73.6	84.2	70.7	74.9	0.742
12	76.0	80.5	47.8	18.3	69.0	71.4	83.6	70.2	73.9	0.733
13	76.0	81.5	37.0	28.3	70.2	70.1	87.7	66.9	74.0	0.737
14	76.3	77.9	32.8	31.7	68.5	71.6	85.9	67.4	72.8	0.728
15	76.3	80.7	36.4	33.3	70.2	69.7	86.7	68.5	73.9	0.737
16	76.1	78.7	32.1	0.3	69.4	70.7	84.5	69.1	73.1	0.730

Tabla 5.11: Resultados Decision Tree - Etapa Detección.
Fuente: Elaboración propia.

Al igual que en Naive Bayes, este algoritmos de Random Forest muestra una mejora de rendimiento al conservar las Stopwords, se aumenta el nivel de accuracy en cerca de 1 %. Por el lado de

los N-grams, se puede apreciar claramente que utilizar Bi-grams mejora los resultados en alrededor de 1 %. No se logra ver claramente si la variable *Polaridad* tiene un efecto positivo o negativo, en algunos casos mejora los indicadores, pero en otros lo empeora, por lo que no se puede concluir con respecto a la inclusión de esta variable. Finalmente, realizar Stemming mejora los resultados del clasificador.

De todas las configuraciones de preprocesamiento de texto analizadas para el algoritmo de Decision Tree, la que mejor desempeño obtuvo fue la número 14, donde no fueron removidas las Stopwords, no se realizó Stemming, se incorporó la Polaridad y se utilizaron Uni-grams. Este modelo presenta el nivel de *Acurracy* (74.9) y *F-Measure* (0.743) más altos, además en la clase "Reclamo" presenta el mejor valor de *Precision* (77.2).

Con el fin de apreciar de mejor forma los resultados obtenidos en esta configuración, la tabla 5.12 muestra la matriz de confusión para este escenario.

		Clase predicha				
		Reclamo	Otro	Consulta	Respuesta	Recall
clase real	Reclamo	652	9	133	15	80.6 %
	Otro	40	10	9	1	16.7 %
	Consulta	122	3	385	8	74.3 %
	Respuesta	31	2	21	127	70.2 %
	Precision	77.2 %	41.7 %	70.3 %	84.1 %	

Tabla 5.12: Matriz de Confusión - Etapa de Detección DT - Configuración N^{a}10.
Fuente: Elaboración propia.

De la matriz de confusión se puede extraer que el clasificador es más asertivo para la clase *Respuesta*, seguido de *Reclamo*, *Consulta* y *Otro* respectivamente. Aunque considerando la medida F-Measure para cada clase, la que mejor se encuentra es la clase *Reclamo*, la más importante del problema. Además se puede apreciar que el clasificador se equivoca en las predicciones de forma pareja, es decir, proporcional a la cantidad de datos en cada una de las otras clases.

5.2.2. Resumen y Elección

Habiendo analizado los cuatro algoritmos propuestos, se debe escoger el que mejor desempeño posea. En la tabla 5.13 se muestra el mejor resultado de cada uno de los algoritmos.

Quien mejor nivel de Accuracy y F-Measure alcanza es Support Vector Machines, seguido de Random Forest, Decision Trees y Naive Bayes. Analizando las clases, se puede apreciar que SVM es quien tiene mejor *Precision* en la clase de mayor importancia.

En conclusión el modelo escogido finalmente en la Etapa de Detección es Support Vector Machines, en cuya configuración de preprocesamiento no se removieron las Stopwords, se realizo Stemming, no se incorporó la polaridad del tweet y se utilizaron Bi-grams.

Algoritmo		Clases				Acc.	F1
		Reclamo	Otro	Consulta	Respuesta		
Naive Bayes	P	79.4	36.8	71.6	64.9	73.0	0.731
	R	78.0	46.7	67.2	75.7		
	F1	78.7	41.2	69.3	69.9		
Support Vector Machines	P	80.9	39.2	76.4	85.1	77.9	0.780
	R	83.8	48.3	73.0	75.7		
	F1	82.3	43.3	74.4	80.1		
Random Forest	P	73.7	73.3	84.4	93.2	77.8	0.768
	R	94.7	36.7	59.5	68.5		
	F1	82.9	48.9	69.8	79.0		
Decision Tree	P	77.2	41.7	70.3	84.1	74.9	0.743
	R	80.6	16.7	74.3	70.2		
	F1	78.9	23.8	72.2	76.5		

Tabla 5.13: Resumen mejores resultados - Etapa Categorización.
Fuente: Elaboración propia.

5.2.3. Datos Ambiguos

Recordando la selección de los datos en el capítulo IV, los tweets ambiguos (tweets donde los etiquetadores se encontraban en desacuerdo) fueron dejados a un lado para el entrenamiento de los algoritmos, pero se indicó que estos debían ser evaluados, ya que representan una porción de los datos reales que se encuentran en Twitter, por ende no se pueden descartar y tienen que ser evaluados para ver como se comporta el clasificador en estos datos ambiguos.

Para evaluar estos tweets ambiguos se tomaron ambas clases como positivas, es decir, si el Etiquetador 1 escogió la clase *Reclamo* y el Etiquetador 2 escogió la clase *Consulta*, ambas opciones serán correctas para el tweet. Por lo tanto si el modelo clasifica el tweet en alguna se estas 2 clases, se tomará como un *Verdadero Positivo*. En esta Etapa de Detección, 266 fueron los tweets descartados por desacuerdo de los etiquetadores.

En la tabla 5.14 se visualiza el desempeño del clasificador construido en los tweets ambiguos según el indicador *Precision*. No es posible calcular las otras métricas de *Recall* y *F-Measure*, debido a que la etiqueta real de los datos cambió y ahora no es única.

Clase	Precision
1	97.7 %
2	71.4 %
3	93.9 %
4	81.2 %
Prom. Pond.	93.8 %

Tabla 5.14: Desempeño en datos ambiguos - Etapa de Detección.
Fuente: Elaboración propia.

A simple vista se puede apreciar que el desempeño en estos datos ambiguos es mucho mejor que en la evaluación de los algoritmos, esto se debe principalmente a que ahora el clasificador posee un margen de error mayor, es decir, tiene más alternativas en donde acertar.

El nivel de *Accuracy* del clasificador en estos datos abiguos es de 93.6 %. De todas formas, estos resultados no tienen el objetivo de analizar que tan bueno es el clasificador, si no que tienen el fin de detectar y observar si está dentro de los rangos permitidos, en otras palabras se busca identificar si los resultados que entrega el clasificador van en la misma linea de lo obtenido en la evaluación del algoritmo. Tal como se ve, los resultados obtenidos en estos datos ambiguos son mejores a lo obtenido en el entrenamiento del modelo, el clasificador construido se desempeña bien y entrega predicciones de acuerdo a lo esperado.

5.3. Categorización de Reclamos

El segundo problema a resolver consiste en identificar a que categoría(s) pertenece una opinión de Twitter. Tal como se indicó en el capítulo anterior, las clases para esta etapa son 9, las primeras 8 tienen relación con los servicios que se ofrecen en el rubro de las Telecomunicaciones en Chile, mientras que la última indica que el mensaje no pertenece a ninguna de estas.

Es necesario destacar que existe una diferencia crucial entre esta etapa y la anterior, para esta caso las categorías son "No Excluyentes", esto quiere decir que un tweet puede pertenecer a una o más categorías a la vez. En consecuencia no se puede utilizar el mismo método que antes, ahora se debe desarrollar un modelo que pueda clasificar en más de una clase a una muestra a la vez. Para resolver esto se plantea construir un algoritmo binario para cada una de las 8 primeras clases, que permita determinar de manera independiente si un tweet pertenece o no a una clase determinada. La figura 5.2 muestra el proceso que sigue el modelo a construir.

La explicación de este modelo es como sigue: se procesa el tweet con cada uno de los 8 algoritmos correspondientes a cada clase, de ser positivo el resultado (si pertenece a dicha clase), el tweet es categorizado en esa clase. En el caso que los 8 algoritmos entreguen como resultado negativo (no pertenece a dicha clase), el tweet es categorizado en la clase 9 (Ninguna de las anteriores). Si algún algoritmo entrega como resultado negativo, pero existe al menos uno que entregó positivo, el tweet no es clasificado como la clase 9. Se clasifica en esta última si y solo si los 8 algoritmos entregaron como resultado negativo.

5.3.1. Modelo Bag-Of-Words

Al igual que en la Etapa de Detección, para trabajar con texto se deben aplicar técnicas de preprocesamiento para construir una estructura entendible por los algoritmos de Machine Learning. El flujo de operaciones que se le realizan al texto en esta Etapa de Categorización es similar al realizado antes, en la figura 5.1 se muestran las distintas operaciones realizadas al texto. La única diferencia con respecto a la Etapa de Detección, es que ahora no se incluye la variable "Polaridad", ya que no se tiene ningún supuesto con referencia a las clases para esta variable. En consecuencia, se

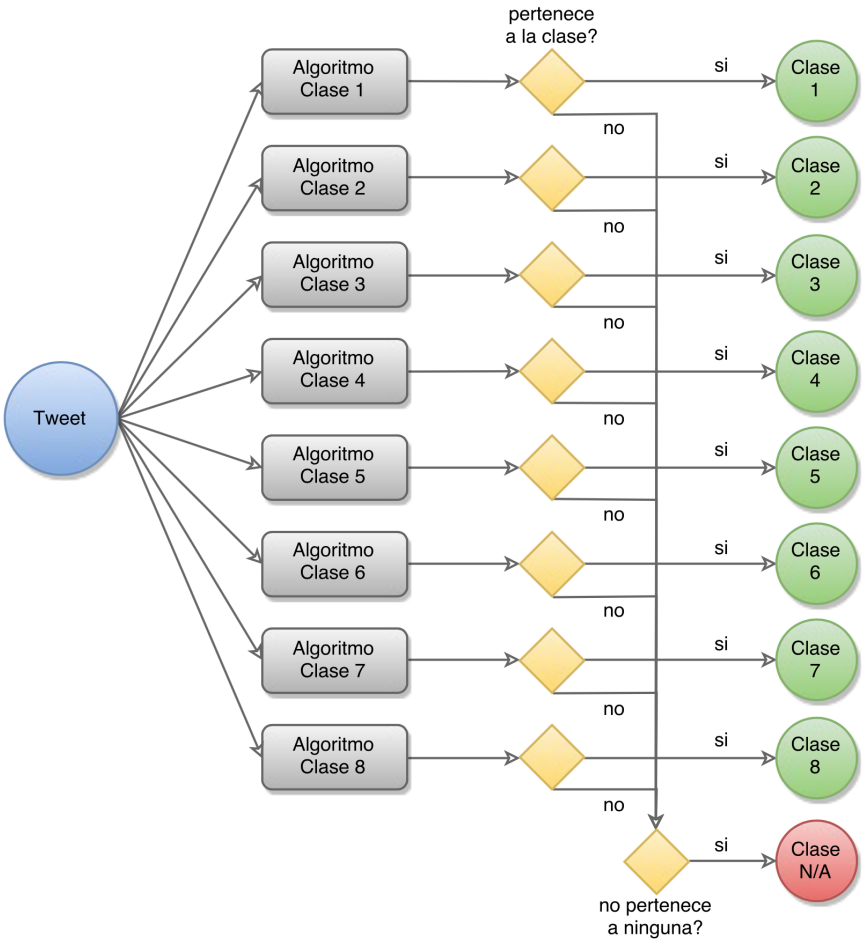

Figura 5.2: Modelo de clasificación - Etapa de Categorización.
Elaboración propia.

realizan tan solo 8 combinaciones o configuraciones de preprocesado de texto, las cuales se detallan en la tabla 5.15. Estas combinaciones consisten en variar la *Remoción de Stopwords*, realización de *Stemming* y utilización de *Uni-grams* o *Bi-grams*.

De igual manera que en la Etapa de Detección los algoritmos evaluados son cuatro, con la salvedad que ahora son utilizados en su variante binaria y no multi-clases, que fue lo utilizado en la Etapa de Detección.

1. Naive Bayes
2. Support Vector Machines
3. Random Forest
4. Decision Tress.

En lo que sigue se exponen los resultados para cada uno de los 4 algoritmos de Machine Learning evaluados. Es necesario aclarar que ahora se construye un clasificador independiente para cada clase, por lo que no se muestran las métricas generales de *Accuracy* y *F-Measure* para cada confi-

Configuración	Remoción de Stopwords	Stemming	Bi-grams
1	✓	✓	✓
2	✓	✓	-
3	✓	-	✓
4	✓	-	-
5	-	✓	✓
6	-	✓	-
7	-	-	✓
8	-	-	-

Tabla 5.15: Configuraciones de preprocesamiento de texto - Etapa de Categorización. Fuente: Elaboración propia.

guración. Tampoco se muestra la matriz de confusión, ya que se escogerá un algoritmo para clase y no uno para todas, Por lo tanto el resultado de esta matriz no tiene influencia en la decisión final.

Cada algoritmo fue evaluado utilizando 5-Fold-cross-validation, esto es entrenar el modelo con el 80% de los datos y testearlo con el 20% restante, y como regla para decidir que configuración es la que mejor rendimiento entrega se considera el valor de los indicadores de cada clases en el siguiente orden:

1. F-Measure
2. Precision
3. Recall

5.3.1.1. Naive Bayes

El primer algoritmo evaluado es Naive Bayes en su versión Simple-Class. Este algoritmo no presenta opciones para variar sus parámetros, puesto que es un modelo que se basa en probabilidades condicionales, las cuales no tienen parámetros.

La tabla 5.16 muestra los resultados obtenidos para cada una de los 8 combinaciones de preprocesamiento de texto al aplicar el algoritmo de Naive Bayes de la librería WEKA. Esta tabla muestra el nivel de *Precision*, *Recall* y *F-Measure* para la clase positiva, estas representan las métricas para que el tweet pertenezca a la clase indicada.

De los resultados obtenidos tras analizar las diferentes configuraciones de preprocesado de texto se puede concluir que la utilización de Uni-grams es mejor que Bi-grams en prácticamente todas las clases. En segundo lugar, la remoción de Stopwords también aumenta el desempeño del algoritmo en todas las clases. Finalmente, con respecto a la incorporación de Stemming no se puede deducir nada, ya que existen casos en donde mejora y otros en donde empeoran los indicadores.

En la tabla 5.16 se pueden observar en color celeste las configuraciones de preprocesamiento

Config.		Clases							
		1	2	3	4	5	6	7	8
1	P	43.8	28.3	72.1	35.8	49.6	35.6	31.9	22.5
	R	71.4	70.9	86.5	61.2	66.2	68.9	65.7	37.5
	F1	0.543	0.404	0.786	0.452	0.567	0.470	0.430	0.281
2	P	48.1	33.1	72.1	38.7	58.4	40.5	35.2	23.1
	R	74.4	68.4	85.3	63.1	65.2	71.1	72.9	50.0
	F1	0.584	0.446	0.782	0.480	0.616	0.516	0.474	0.316
3	P	46.2	31.7	73.2	38.9	52.8	27.7	33.3	29.6
	R	69.0	75.9	85.1	59.2	65.7	48.9	61.4	33.3
	F1	0.553	0.448	0.787	0.469	0.568	0.353	0.432	0.314
4	P	49.6	36.6	73.4	42.1	61.0	37.3	37.2	28.9
	R	69.0	75.9	85.8	59.2	67.2	58.9	68.6	45.8
	F1	0.577	0.494	0.791	0.492	0.639	0.457	0.482	0.355
5	P	40.7	24.1	68.4	31.0	41.9	31.3	31.5	18.0
	R	71.9	70.9	84.0	69.9	63.1	67.8	72.9	37.5
	F1	0.520	0.360	0.754	0.430	0.504	0.428	0.440	0.243
6	P	42.9	30.6	67.6	37.4	49.6	35.5	31.0	18.9
	R	69.0	67.1	81.2	65.0	65.7	65.6	6.86	41.7
	F1	0.529	0.421	0.738	0.475	0.565	0.459	0.427	0.260
7	P	40.3	26.1	68.8	29.4	41.7	24.8	34.0	0.225
	R	69.5	72.2	83.3	65.0	63.6	57.8	70.0	0.375
	F1	0.510	0.384	0.754	0.405	0.504	0.347	0.458	0.281
8	P	42.5	31.4	67.8	36.9	50.8	30.1	34.8	24.4
	R	66.0	67.1	80.5	56.3	64.1	57.8	68.6	45.8
	F1	0.517	0.427	0.736	0.446	0.567	0.395	0.462	0.319

Tabla 5.16: Resultados Naive Bayes - Etapa Categorización.
Fuente: Elaboración propia.

que obtienen los mejores resultados para cada clase. Para las clases 1 y 6 el mejor resultado se tuvo cuando se removieron la Stopwords, se realizó Stemming y se consideraron Uni-grams. En cambio, para las otras 6 clases el mejor resultado se tuvo cuando se removieron las Stopwords, no se realizó Stemming y se consideraron Uni-grams.

5.3.1.2. Suport Vector Machines

El segundo algoritmo evaluado fue Support Vector Machines en su versión Simple-Class. Como se mencionó anteriormente, se utilizaron los parámetros por defectos de la librería WEKA, para el caso de SVM detallan en la tabla 5.4. Al igual que en el caso anterior, se evaluó el desempeño de este algoritmo para cada una de las 8 conifguraciones de preprocesamiento de texto. La tabla 5.5 muestra los resultados obtenidos tras aplicar el algoritmo de SVM de la librería WEKA. Se muestra el nivel de *Precision*, *Recall* y *F-Measure* para la clase positiva.

Config.		Clases							
		1	2	3	4	5	6	7	8
1	P	87.9	89.6	91.9	86.0	90.8	85.3	88.9	100.0
	R	46.3	54.4	84.7	47.6	60.1	32.2	45.7	16.7
	F1	0.606	0.677	0.881	0.613	0.723	0.468	0.604	0.286
2	P	78.5	75.0	90.5	74.7	85.2	75.7	75.5	81.8
	R	63.1	53.2	88.8	54.4	75.8	58.9	52.9	37.5
	F1	0.699	0.622	89.7	0.629	0.802	0.663	0.622	0.514
3	P	92.6	88.6	91.3	84.2	97.8	91.3	96.9	100.0
	R	42.9	49.4	85.8	46.6	67.2	23.3	44.3	12.5
	F1	0.586	63.4	0.885	0.600	0.796	0.372	0.608	0.222
4	P	76.8	74.2	91.7	75.0	95.3	73.9	87.0	87.5
	R	57.1	58.2	89.5	52.4	82.8	37.8	57.1	29.2
	F1	0.655	0.652	0.906	61.7	0.886	0.500	0.690	0.438
5	P	83.9	95.2	91.7	86.5	90.4	90.3	93.5	100.0
	R	48.8	50.6	82.1	43.7	52.5	31.1	41.4	12.5
	F1	0.617	0.661	0.866	0.581	0.665	0.463	0.574	0.222
6	P	73.7	77.2	90.5	67.9	87.1	71.2	78.3	63.6
	R	60.6	55.7	88.1	53.4	71.7	52.2	51.4	29.2
	F1	0.665	0.647	0.893	0.598	0.787	0.603	0.621	0.400
7	P	82.4	92.5	92.1	87.5	95.8	91.3	96.7	100.0
	R	43.8	46.8	83.5	40.8	58.1	23.3	41.4	12.5
	F1	0.572	0.622	87.6	0.556	0.723	0.372	0.580	0.222
8	P	74.7	75.8	91.3	71.2	94.0	71.4	89.4	80.0
	R	58.1	59.5	87.7	50.5	79.8	38.9	60.0	33.3
	F1	0.654	0.667	89.4	0.591	0.863	0.504	0.718	0.471

Tabla 5.17: Resultados Support Vector Machines - Etapa Categorización.
Fuente: Elaboración propia.

De los resultados obtenidos tras analizar las diferentes configuraciones de preprocesado de texto se puede concluir que la utilización de Uni-grams es mejor que Bi-grams en prácticamente todas las clases. En segundo lugar, con respecto a la remoción de Stopwords no es posible deducir nada, existen casos donde mejora u otros donde empeoran los indicadores. Finalmente, con respecto a la incorporación de Stemming en una gran parte de los casos mejora el rendimiento, sin embargo no es siempre igual.

En la tabla 5.17 se pueden observar en color celeste las configuraciones de preprocesamiento que obtienen los mejores resultados para cada clase. Las clases 1,4,6 y 8 tiene su mejor rendimiento en la configuración N°2. La clase 2 tiene su mejor desempeño en la configuración N°1. Las clases 3 y 4 tiene su mejor rendimiento en la configuración N°4. Y finalmente, la clase 7 tiene su mejor rendimiento en la configuración N°8.

5.3.1.3. Random Forest

El tercer algoritmo a evaluar es Random Forest en su versión Simple-Class. Como se mencionó anteriormente, se utilizaron los parámetros por defectos de la librería WEKA, para el caso de Random Forest estos se detallan en la tabla 5.7. Al igual que en el caso anterior se evaluó el desempeño de este clasificador en las 8 configuraciones de preprocesamiento de texto. La tabla 5.18 visualiza los resultados.

Config.		Clases							
		1	2	3	4	5	6	7	8
1	P	100	0	94.5	100	100	100	0	100
	R	2.5	0	59.5	1.0	5.1	2.2	0	8.3
	F1	0.048	0	0.730	0.019	0.096	0.043	0	0.154
2	P	100	100	94.7	100	100	100	0	100
	R	13.3	1.3	74.2	6.8	27.8	8.9	0	8.3
	F1	0.235	0.025	0.832	0.127	0.435	0.163	0	0.154
3	P	100	0	94.7	100	100	100	0	100
	R	2.0	0	57.7	1.0	9.6	2.2	0	8.3
	F1	0.039	0	0.717	0.019	0.175	0.043	0	0.154
4	P	100	100	94.2	100	100	100	0	100
	R	8.9	5.1	75.3	0.039	26.8	3.3	0	8.3
	F1	0.163	0.086	0.837	0.075	0.422	0.065	0	0.154
5	P	100	100	96.4	0	100	100	0	100
	R	2.0	1.3	49.5	0	3.5	2.2	0	8.3
	F1	0.039	0.025	0.654	0	0.068	0.043	0	0.154
6	P	100	100	94.4	0	100	100	0	100
	R	4.9	3.8	63.3	0	9.6	2.2	0	8.3
	F1	0.094	0.073	0.758	0	0.175	0.043	0	0.154
7	P	100	100	98.0	100	100	100	0	100
	R	2.5	1.3	46.7	1.0	5.1	2.2	0	8.3
	F1	0.048	0.025	0.633	0.019	0.096	0.043	0	0.154
8	P	100	100	96.9	0	100	1	0	100
	R	4.4	2.5	57.4	0	8.6	2.2	0	8.3
	F1	0.085	0.049	0.721	0	0.158	0.043	0	0.154

Tabla 5.18: Resultados Random Forest - Etapa Categorización.
Fuente: Elaboración propia.

A simple vista los resultado de este algoritmo parecen malos para todas las clases a excepción de la 3, esto se debe a que es la que posee la mayor cantidad de datos y el algoritmo tiene más muestras para aprender, sin embargo para las otras clases en donde la clase positiva y negativa poseen una disparidad muy grande en cuanto al tamaño de datos que lo conforman (90 % vs 10 %), el algoritmo no es capaz de captar la variabilidad en los datos, por esto en muchos casos la *Precision* alcanza la perfección, pero el *Recall* es muy pobre. Esto indica que tan solo aprende de los términos más representativos, pero no los ambiguos o menos frecuentes.

Con respecto al análisis de que técnica de preprocesamiento de texto es mejor para este algoritmo, no es posible concluir mucho, debido a que el rendimiento en las distintas clases es muy bajo, además no existe una correlación clara que indique que una técnica es mejor que otra.

5.3.1.4. Decision Trees

Como último algoritmo a analizar se tuvo a Decision Trees en su versión Simple-Class. Como se mencionó anteriormente, se utilizaron los parámetros por defectos de la librería WEKA, para el caso de Random Forest estos se detallan en la tabla 5.10. Al igual que en el caso anterior se evaluó el desempeño de este clasificador en las 8 configuraciones de preprocesamiento de texto. La tabla 5.19 visualiza los resultados.

Config.		Clases							
		1	2	3	4	5	6	7	8
1	P	66.4	81.1	90.5	68.6	90.6	69.8	70.6	100
	R	39.9	54.4	82.1	46.6	53.5	41.1	51.4	8.3
	F1	0.498	0.652	0.861	0.555	0.673	0.517	0.595	0.154
2	P	68.6	83.9	90.5	69.1	87.4	69.1	73.6	100
	R	39.9	59.5	82.3	45.6	56.1	42.2	55.7	8.3
	F1	0.505	0.696	0.862	0.550	0.683	0.524	0.634	0.154
3	P	67.7	91.7	90.2	77.3	87.5	62.5	68.9	0
	R	33.0	55.7	81.6	49.5	49.5	16.7	44.3	0
	F1	0.444	0.693	0.857	0.604	0.632	0.263	0.539	0
4	P	71.3	91.8	89.8	78.9	92.1	69.6	70.2	0
	R	35.5	57.0	82.1	54.4	47.0	17.8	47.1	0
	F1	0.474	0.703	0.858	0.644	0.622	0.283	0.564	0
5	P	53.8	69.6	86.6	69.3	85.7	67.9	66.7	100
	R	37.9	49.4	81.4	50.5	48.5	40.0	48.6	8.3
	F1	0.445	0.578	0.839	0.584	0.619	0.503	0.562	0.154
6	P	54.9	77.6	87.8	68.5	82.8	66.7	67.3	100
	R	38.4	48.1	80.0	48.5	53.5	37.8	52.9	8.3
	F1	0.452	0.594	0.837	0.568	0.650	0.482	0.592	0.154
7	P	54.1	84.0	89.3	74.6	87.9	56.5	63.3	0
	R	29.6	53.2	81.2	45.6	47.5	14.4	44.3	0
	F1	0.382	0.651	0.850	0.566	0.616	0.230	0.521	0
8	P	59.6	80.8	86.6	80.3	90.4	55.6	64.7	0
	R	30.5	53.2	79.5	51.5	47.5	11.1	47.1	0
	F1	0.404	0.641	0.829	0.627	0.623	0.185	0.545	0

Tabla 5.19: Resultados Decision Tree - Etapa Categorización.
Fuente: Elaboración propia.

De los resultados obtenidos tras analizar las diferentes configuraciones de preprocesado de texto se puede concluir que la utilización de Uni-grams es mejor que Bi-grams en prácticamente todas las

clases. En segundo lugar, la remoción de Stopwords también aumenta el desempeño del algoritmo en todas las clases. Finalmente, con respecto a la incorporación de Stemming no se puede deducir nada, ya que existen casos en donde mejora y otros en donde empeoran los indicadores.

En la tabla 5.19 se pueden observar en color celeste las configuraciones de preprocesamiento que obtienen los mejores resultados para cada clase. Para las clases 1, 3, 5, 6 y 7 el mejor resultado se tuvo cuando se removieron la Stopwords, se realizó Stemming y se consideraron Uni-grams. En cambio, para las clases 2 y 4 el mejor resultado se obtuvo cuando se removieron las Stopwords, no se realizó Stemming y se consideraron Uni-grams.

5.3.2. Resumen y Elección

Habiendo analizado todos los algoritmos propuestos, se debe escoger el que mejor desempeño posea. La tabla 5.20 muestra el mejor resultado de cada clase en cada uno de los 4 algoritmos vistos. Para este caso, a diferencia de la Etapa de Detección, se escogerá el mejor algoritmo para cada clase, dado que son modelos independientes. Del mismo modo que antes, se escoge el modelo que posea el mejor indicador *F-Measure*, seguido de *Precision* y *Recall*.

Algoritmo.		Clases							
		1	2	3	4	5	6	7	8
Naive Bayes	P	48.1	36.6	73.4	42.1	61.0	40.5	37.2	28.9
	R	74.4	75.9	85.8	59.2	67.2	71.1	68.6	45.8
	F1	0.584	0.494	0.791	0.492	0.639	0.516	0.482	0.355
	Conf.	2	4	4	4	4	2	4	4
Support Vector Machines	P	78.5	89.6	91.7	74.7	95.3	75.7	89.4	81.8
	R	63.1	54.4	89.5	54.4	82.8	58.9	60.0	37.5
	F1	0.699	0.677	0.906	0.629	0.886	0.663	0.718	0.514
	Conf.	2	1	4	2	4	2	8	2
Random Forest	P	100	100	94.2	100	100	100	0	100
	R	13.3	5.1	75.3	6.8	27.8	8.9	0	8.3
	F1	0.235	0.086	0.837	0.127	0.435	0.163	0	0.154
	Conf.	2	3	3	2	2	2	-	-
Decision Tress	P	68.6	91.8	90.5	78.9	87.4	69.1	73.6	100
	R	39.9	57.0	82.3	54.4	56.1	42.2	55.7	8.3
	F1	0.505	0.703	0.862	0.644	0.683	0.524	0.634	0.154
	Conf.	2	4	2	4	2	2	2	-

Tabla 5.20: Resumen mejores resultados - Etapa Categorización.
Fuente: Elaboración propia.

Como resumen general al comparar los cuatro algoritmos de Machine Learning analizados se puede observar que Random Forest es quien entrega los peores resultados con un promedio de 0.255 en el indicador F1. En tercer lugar se encuentra Naive Bayes con resultados promedio de 0.544. En segundo lugar se encuentra Decision Tress con resultados promedio de 0.589. En primer lugar se encuentra SVM con un promedio de 0.712 en el indicador F1.

Finalmente, a nivel de clases se escogieron los algoritmos que mejor resultados obtuvieron, para las clases 1, 3, 5, 6, 7 y 8 quien mejor desempeño entrega es Support Vector Machines, mientras que para la clase 2 y 4 el algoritmo Decision Trees entrega mejores resultados. Un dato interesante es que de las configuraciones finalmente escogidas, todas ellas utilizan Uni-grams.

Con el modelo ya construido es posible evaluar como se comporta el clasificador en la clase número 9 (ninguna de las anteriores), y además calcular métricas generales del clasificador, como *Accuracy*, *Precision* promedio, *Recall* promedio y *F-Measure* promedio. La tabla 5.21 muestra estos resultados.

Clase	Precision	Recall	F-Measure
1	78.5	63.1	0.700
2	91.8	57.0	0.703
3	91.7	89.5	0.906
4	78.9	54.4	0.644
5	95.3	82.8	0.883
6	75.7	58.9	0.663
7	89.4	60.0	0.718
8	81.8	37.5	0.514
9	90.8	99.8	0.951
Prom. pond.	87.8	80.2	0.838

Tabla 5.21: Desempeño final modelo desarrollado - Etapa de Categorización.
Fuente: Elaboración propia.

Como principal resultado del clasificador se tiene que tiene la capacidad de clasificar 4 de cada 5 tweets de forma correcta. Esto se refleja en un nivel de Accuracy de 81.3 %. Además, se puede observar que todas las clases poseen una *Precision* superior al 75 %, es decir, en la peor clase el clasificador acierta 3 de cada 4 tweets.

5.3.3. Datos Ambiguos

De igual forma que lo realizado en la Etapa de Detección, es necesario evaluar el clasificador construido en los tweets ambiguos. Siguiendo el mismo análisis visto antes, se tomarán ambas elecciones de los etiquetadores como verdaderas, con la salvedad que para este caso al ser no excluyentes las opciones, podrían existen casos donde hayan más de 2 clases posibles.

Para la Etapa de Categorización, 262 fueron los tweets descartados por desacuerdo entre los etiquetadores. La tabla 5.22 muestra el desempeño del clasificador según el indicador de *Precision* para las 9 clases en estos tweets ambiguos. No es posible calcular medidas de *Recall* ni *F-Measure*, ya que la etiqueta real de los datos no es única.

El nivel de *Accuracy* del clasificador en estos datos ambiguos es de 82.1 %., sin embargo, el objetivo de analizar los tweets ambiguos es para comprobar que el modelo construido entregue resultados acorde a los esperado. Tal como se observa, los resultados son similares a los obtenidos

Clase	Precision
1	85.7
2	75.0
3	94.4
4	86.7
5	91.3
6	100
7	77.8
8	75.0
9	69.2
Prom. Pond.	88.4

Tabla 5.22: Desempeño en datos ambiguos - Etapa de Categorización.
Fuente: Elaboración propia.

con los datos de entrenamiento, por lo que se espera que el clasificador se comporte de igual manera en la realidad frente a estos tweets ambiguos.

Capítulo 6

Diseño e Integración del Módulo de Reclamos

El presente capítulo tiene por objetivo diseñar e implementar el módulo funcional de reclamos en la plataforma web OpinionZoom. La primera parte consiste en desarrollar la arquitectura necesaria para implementar el algoritmo de clasificación, y en una segunda parte se tiene el diseñar el módulo web e integrarlo en el sitio de OpinionZoom.

6.1. Clasificador de reclamos

En primer lugar, luego de haber determinado cual es el mejor modelo de clasificación, se tiene el entrenar el algoritmo respectivo para cada una de las dos etapa: *Detección* y *Categorización* de reclamos. Para llevar a cabo esto, se considera el 100 % de los datos para realizar el entrenamiento final de los algoritmos, es decir, a diferencia de lo realizado para la evaluación de los modelos en el capítulo 5, en donde se utilizaba tan solo el 80 % de los datos, ahora se utiliza todo el set de datos.

Finalmente el "output" de modelo de clasificación es un archivo *.model*[1] el cual almacena toda la lógica del clasificador. Recordando lo expuesto en el capítulo anterior, la primera etapa correspondiente a la detección de reclamo considera tan solo 1 clasificador multi-clase capaz de escoger entre 4 clases, por ende se tiene un archivo denominado *det.model*, sin embargo, para la etapa de categorización se diseñó y construyó un modelo que contempla a 8 clasificadores, por lo que para esta se tienen 8 archivos: *cat_i.model* donde *i = (1,...,8)*.

6.1.1. API de reclamos (RAPI)

Teniendo los archivos de los modelos ya construidos, se tuvo como siguiente paso el construir el clasificador, el cual se encapsuló en una clase en java con el objetivo de poder ser integrado y

[1] archivo de texto serializado, el cual es interpretado por la librería WEKA.

utilizado en el sistema de OpinonZoom.

En detalle, la API de reclamos posee dos métodos con los cuales interactuar:

- *classifyDet (texto)*
- *classifyCat (texto)*

Ambos reciben como "input" un tweet, la diferencia radica en el "output", el primero entrega la categoría a la que pertenece referente a la primera etapa, mientras que el segundo método entrega un arreglo con 9 elementos donde cada uno corresponde a la pertenencia o no con respecto a las categorías de la segunda etapa.

Esta API se integró en el sistema se procesamiento de OpinionZoom denominado *OZcore* el cual contempla toda la lógica que existe detrás de la aplicación, aquí es donde se extraen los datos de la base de datos "La Gorda", se procesan, y se almacenan en una base de datos intermedia (Ozelote), que es la encargada finalmente de alimentar el sitio web donde se despliega la información referente a los tweets.

6.2. Integración del clasificador de reclamos

El sistema de OpinionZoom funciona buscando tweets de un conjunto de usuarios chilenos, y que dichos tweets contengan ciertas Keywords específicas. Estos tweets son almacenados en una base de datos implementada en PostgreSQL, denominada "Ozelote", donde se encuentran todos los datos que posteriormente son mostrados en la página web. Para el módulo de reclamos desarrollado tan solo dos tablas de este modelo son necesarias, la tabla *keyword* y la tabla *tweet*. La primera representa a todas las keywords que el sistema busca y la segunda representa la tabla en donde se guardan los tweets que contienen estas keywords.

En la figura 6.1 se muestra el diagrama entidad-relación construido para el módulo de reclamos, este cuenta con dos tablas propias del sistema OpinizonZoom: *tweet* y *keyword*, y cuatro tablas propias: *keysreclamos*, *reclamos*, *deteccion_aggregate* y *categorizacion_aggregate*. Estas nuevas tablas son acopladas al sistema actual de OpinionZoom con el fin de integrar el módulo de forma completa. A continuación se detalla cada tabla con los campos que contiene cada una dentro del modelo diseñado. No se detallan todos los campos de las tablas *tweet* y *keyword*, ya que para el funcionamiento del Módulo de Reclamos no son necesarios todos, solo algunos de ellos.

1. **tweet:** Esta tabla contiene todos los tweets almacenados por la plataforma de OpinionZoom.

    - idtweet: Corresponde al identificador del tweet para la base de datos Ozelote.
    - text: Representa el texto del tweet.
    - retweet: Este campo indica si el tweet en cuestión corresponde a un Retweet, si su valor es distinto de -1, lo es.
    - reply: Este campo indica si el tweet corresponde a una respuesta de otro. Si su valor es igual a 0, no lo es.
    - tiempostamp: corresponde a la fecha de emisión del tweet.

2. **keyword:** Esta tabla almacena todas la keywords que son buscadas por el sistema de OpinionZoom en Twitter.

    - idkeyword: Corresponde al identificador del tweet para la base de datos Ozelote.
    - trackeable: Representa el valor (texto) de la keyword.

3. **keysreclamos:** Esta tabla tiene la finalidad de almacenar las keywords para realizar el análisis de reclamos. Para lo desarrollado en esta memoria, estas keywords serían 4: @AyudaMovistarCL, @entel_ayuda, @VTRSoporte y @miclaro_cl.

    - keyword_idkeyword: Llave foránea que referencia a la keyword trackeada por el sistema de OpinionZoom.

4. **reclamos:** Esta tabla tiene la finalidad de almacenar los tweets clasificados por el algoritmo construido.

    - texto: Corresponde al texto del tweet.
    - idtwittertweet: Identificador de tweet (según Twitter).
    - timestamp: Fecha de emisión del tweet.
    - deteccion: Corresponde a la clase predicha por el algoritmo en la Etapa de Detección.
    - categorización: Corresponde a la clase predicha por el algoritmo en la Etapa de Categorización.
    - keysreclamos_idkeysreclamos: llave foránea que referencia a la keyword del módulo de reclamos.

5. **deteccion_aggregate:** Esta tabla tiene la finalidad de almacenar datos agregados para una rápida búsqueda y por ende una menor respuesta por parte del sistema de OpinionZoom. Esto se traduce finalmente, en un tiempo menor de carga de la pagina web para el módulo de reclamos.

    - categoria: Corresponde a una de las 4 clases de la Etapa de Detección.
    - ano: Año al cual corresponde la frecuencia.
    - mes: Mes al cual corresponde la frecuencia.
    - dia: Día al cual corresponde la frecuencia.
    - frecuencia: Corresponde a la frecuencia de tweets para esa categoría y en la fecha correspondiente.

6. **deteccion_aggregate:** Esta tabla, al igual que la anterior, tiene la finalidad de almacenar datos agregados para una rápida búsqueda y por ende una menor respuesta por parte del sistema de OpinionZoom, lo que se traduce finalmente, en un tiempo menor de carga de la pagina web para el módulo de reclamos.

    - categoria: Corresponde a una de las 4 clases de la Etapa de Categorización.
    - ano: Año al cual corresponde la frecuencia.
    - mes: Mes al cual corresponde la frecuencia.
    - dia: Día al cual corresponde la frecuencia.

- frecuencia: Corresponde a la frecuencia de tweets para esa categoría y en la fecha correspondiente.

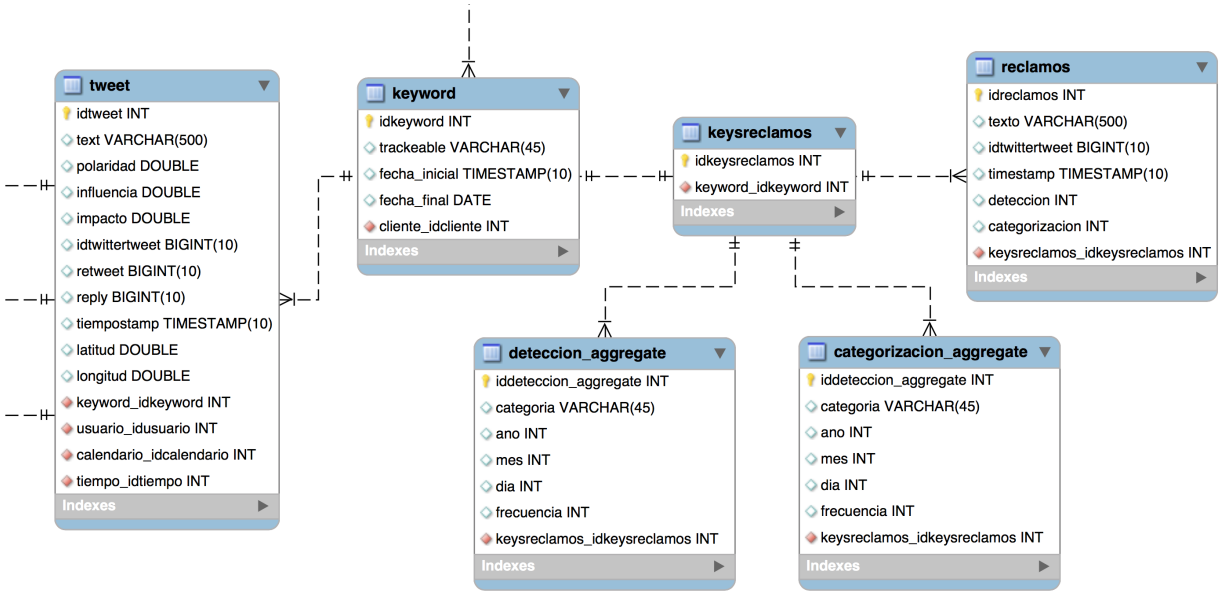

Figura 6.1: Modelo entidad relación - Módulo de Reclamos
Elaboración propia.

6.2.1. Algoritmo de clasificación de tweets

Teniendo listas las tablas en donde se almacenan los datos, se tiene como siguiente paso integrar el clasificador (empaquetado en una clase java) en el sistema de OpinionZoom. Este proceso se lleva a cabo por medio de un algoritmo construido que permite clasificar los tweets de las keywords del módulo y almacenar los resultados en la tabla *reclamos*. A continuación de enumeran y describen los pasos de esta algoritmo implementado.

1. **Selección de keywords:** Se seleccionan todas las keywords de la tabla *keysreclamos*.

2. **Buscar tweets:** Para cada keyword se buscan todos los tweets emitidos en los últimos 30 minutos. Para esto se realiza una consulta sobre la tabla tweets. Además en esta consulta se filtra por la variable *reply*, ya que como se indicó en el capítulo IV, se consideran solo los tweets que no son respuesta a otro, por lo que se consideran tweets cuyo valor de *reply* sea igual a 0.

3. **Clasificar tweets:** Teniendo los tweets identificados, se clasifican de uno en uno con respecto a las dos Etapas: Detección y categorización.

4. **Insertar tweets clasificados:** Los tweets ya clasificados se insertan de uno a la vez en la tabla *reclamos*. Cabe mencionar que para el caso de la categorización, es posible que un mismo tweet haya sido categorizado en más de una clase, por lo que este tweet es insertado tantas veces como clases positivas tenga.

Este algoritmo es ejecutado cada 30 minutos por el sistema de OpinionZoom con el objetivo de no colapsar la plataforma al realizar el procesamiento de los datos de un día en una sola iteración.

6.2.2. Algoritmo de agrupación de tweets

Como se dijo en un comienzo, el módulo de reclamos consta de dos tablas que almacenan datos agregados: *deteccion_aggregate* y *categorizacion_aggregate*, el objetivo de estas tablas es mantener los datos agregados para no tener que realizar operaciones de agrupación en la carga de los datos a la vista de la página web de OpinionZoom.

Para el llenado de estas tablas se realiza un algoritmo para cada Etapa, uno para la Detección y otro para la Categorización, pero ambos siguen la misma lógica detrás. La secuencia de pasos se describe a continuación.

1. **Buscar tweets:** Se seleccionan todos los tweets de la tabla *reclamos* que posean un máximo de un día de antigüedad en su emisión.

2. **Agrupar tweets:** Los tweets seleccionados se agrupan según los campos: categoria, ano, mes, dia y keysreclamos respectivamente. Y se calcula la frecuencia de tweets para cada caso.

3. **Insertar tweets agrupados:** Una vez agrupados los tweets se insertan en lote a la tabla correspondiente.

Este algoritmo es ejecutado 1 vez por día a la 00:05 horas por el sistema de OpinionZoom, esto debido a que los datos son agrupados por día y la medida de fecha más pequeña en la tabla de datos agregados es esta misma, por lo que hacerlo en un intervalo menor no abarcaría el total de datos. por el otro lado, de realizar este proceso en un periodo de tiempo mayor, los datos finalmente mostrados al usuario serian datos muy antiguos y lo que se desea es mostrar los datos más actuales posibles.

6.3. Diseño del Módulo de Reclamos en el sitio OpinionZoom

Como idea principal del Módulo de Reclamos se tiene el exhibir la frecuencia de tweets agrupados por categorías en función del tiempo (ya sea día, mes y/o año).

En primer lugar, es necesario crear el espacio en el menú de navegación del sitio OpinionZoom para poder acceder al módulo de reclamos. La figura 6.2 muestra el panel de navegación de los servicios entregados. En la parte inferior se aprecia como quedó el panel para el módulo de reclamos, este consta de dos apartados: “Reclamos” y “Categorías”. El primero tiene la finalidad de mostrar los resultados correspondiente a la Etapa de Detección, mientras que el segundo muestra los resultados de la Etapa de Categorización.

Se puede apreciar que para cada apartado del módulo de reclamos, se tienen distintas cuentas

(keywords), que para el caso de este trabajo son 4 referentes al rubro de las telecomunicaciones. Esta decisión tuvo como justificación que en OpinionZoom los clientes contratan *keywords* para su análisis, por lo que para el módulo de reclamos se quiso seguir esta misma linea y permitir que los clientes vean el análisis de reclamos por keyword (que en Módulo de Reclamos corresponden a cuenta de Twitter, pero son almacenadas como keywords por el sistema).

El módulo diseñado de esta manera permite solucionar uno de los principales intereses de las empresas, el cual es observar y analizar a la competencia con respecto a los reclamos que ellas reciben. Por lo que una empresa como Movistar podría querer contratar la keyword "@ayuda_entel" con el fin de analizar lo que sucede con esta empresa y viceversa.

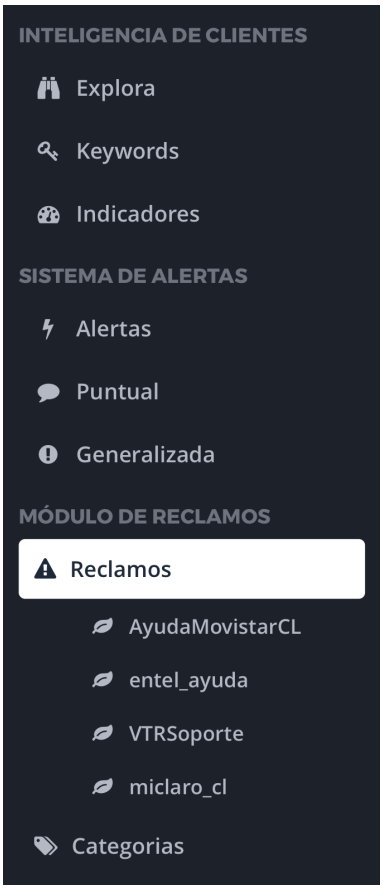

Figura 6.2: Menú de navegación - Sitio web OpinionZoom
Elaboración propia.

En ambas secciones, "Reclamos" y "Categorías", se decidió por exponer la información de manera similar al servicio *Explora* de OpinionZoom. Esta es mostrar la frecuencia de tweets a través del tiempo dentro de un gráfico implementado por medio de la librería ChartJS [69] . Para el caso de los reclamos se decide utilizar una granularidad de día, es decir, el intervalo de tiempo mínimo para agrupar la frecuencia de tweets es por día.

6.3.1. Módulo de Reclamos - Sección Reclamos

En esta primera sección se despliegan los resultados correspondientes a la Etapa de Detección. La figura 6.3 muestra la vista principal de esta sección en el sitio web OpinionZoom para la cuenta (keyword) *@AyudaMovistarCL* en el mes de Diciembre de 2016. En esta vista se tiene un único gráfico que mezcla las 4 categorías posibles en las cuales los tweets son clasificados. Desplazándose sobre el gráfico se puede obtener información detallada de un día en particular, tal como se aprecia en la figura para el día 15 de Diciembre de 2016.

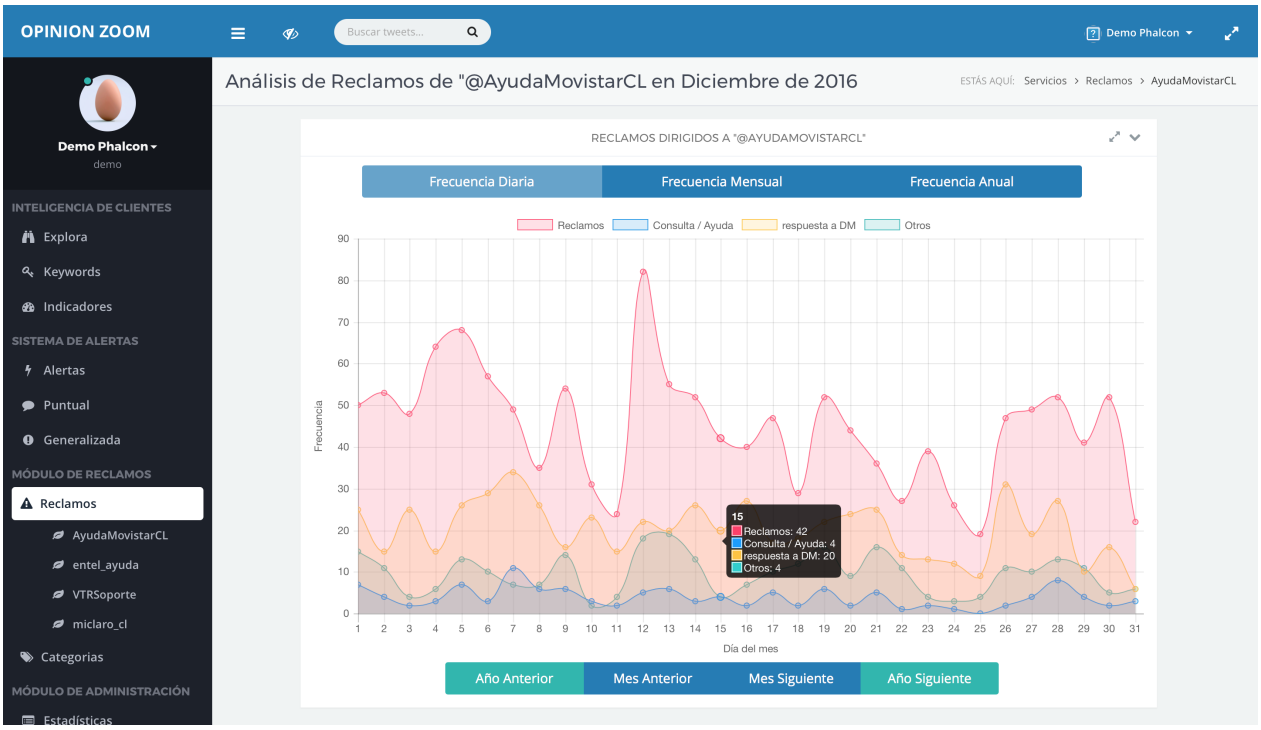

Figura 6.3: Sección Reclamos - Módulo de Reclamos OpinionZoom
Elaboración propia.

A primera vista puede parecer que la información exhibida es un tanto confusa, sin embargo, en la parte superior del gráfico se despliegan las etiquetas de las series. Para esta sección son: *Reclamos*, *consulta / Ayuda*, *Respuesta a DM* y *Otros*. Con la simple acción de *clickear* sobre alguno ellos se puede suprimir dicha serie en el gráfico. En la figura 6.4 se muestra un gráfico en donde solo se despliega la serie *Reclamos*, lo que se llevó a cabo clickeando sobre las otras 3 series. De igual forma para volver a mostrar una serie, basta con clickear nuevamente la etiqueta.

Debajo del gráfico se encuentran 4 botones: *Año Anterior*, *Mes Anterior*, *Mes Siguiente* y *Año Siguiente*. Ellos tienen la finalidad observar los datos en cualquier otro periodo de tiempo. Por medio de estos se puede desplazar hacia el mes anterior o siguiente, y también hacia el año anterior o siguiente (para agilizar la búsqueda a fechas antiguas).

Por otro lado, en la parte superior se tienen 3 botones: “Frecuencia Diaria”, “Frecuencia Mensual” y “Frecuencia Anual”. Con ellos se puede cambiar la información agregada que entrega el gráfico.

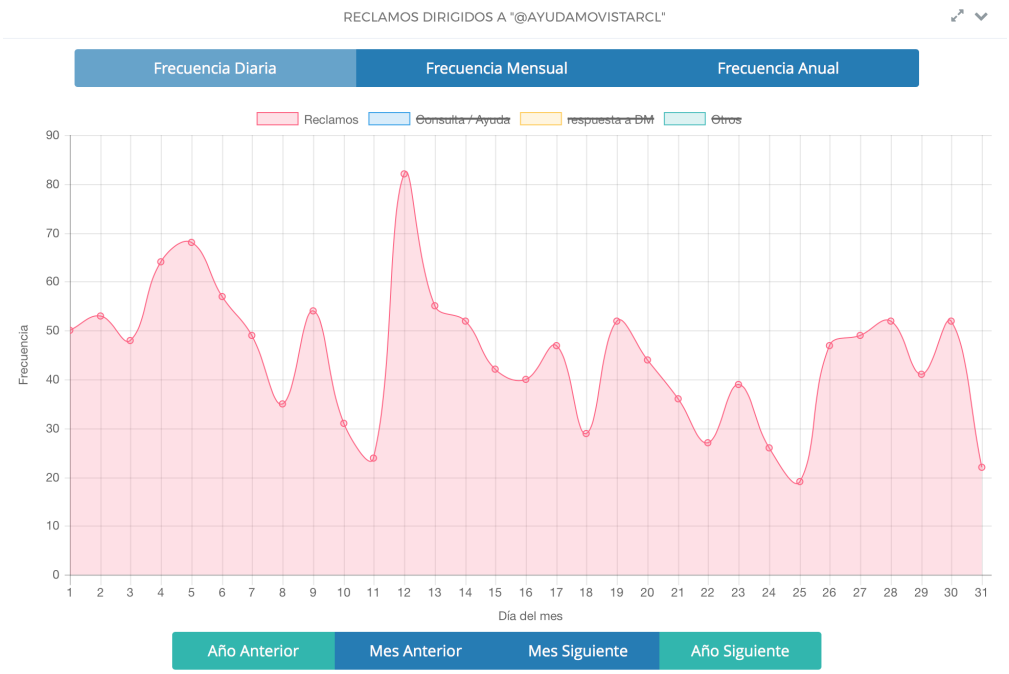

Figura 6.4: Gráfico solo tweets “Reclamos” - Sección Reclamos
Elaboración propia.

El botón “Frecuencia Diaria” muestra la frecuencia diaria, tal como se aprecia en las figuras 6.3 y 6.4.

El botón “Frecuencia Mensual” muestra la frecuencia mensual, en la figura 6.5 se puede apreciar este gráfico. En este caso las opciones para desplazarse a través del tiempo, que se ubican debajo del gráfico, cambian. Ahora solo es posible moverse a o largo de los años.

Finalmente, el botón “Frecuencia Anual” muestra la frecuencia anual, en la figura 6.6 se puede apreciar este gráfico. Para este caso no existen opciones para desplazarse a través del tiempo, esta representa la unidad de tiempo más grande en la cual se muestran los datos.

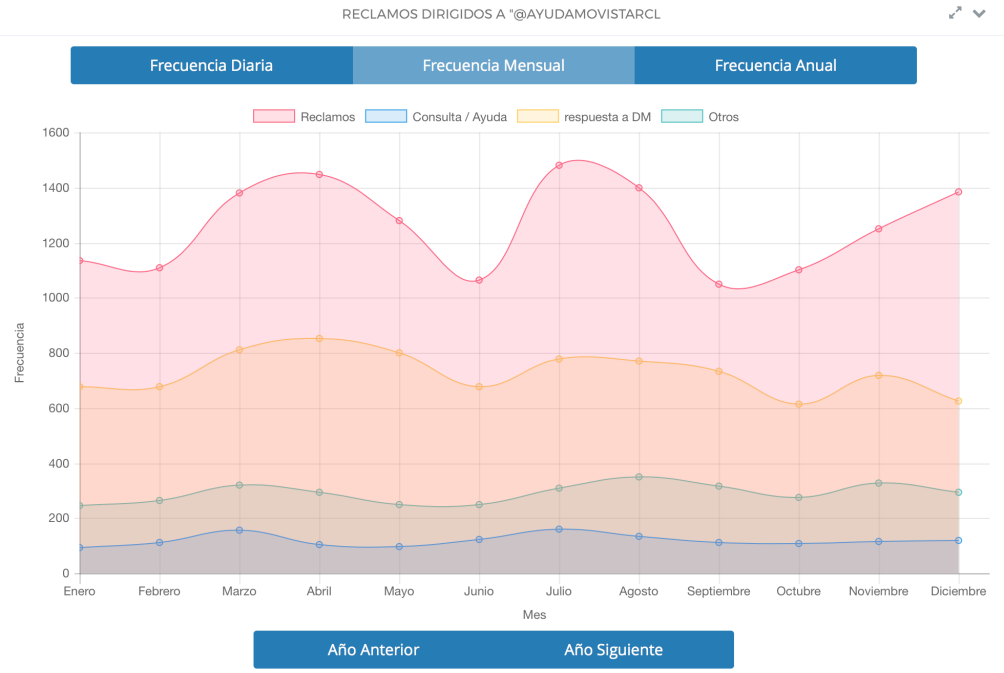

Figura 6.5: Gráfico frecuencia mensual - Sección Reclamos
Elaboración propia.

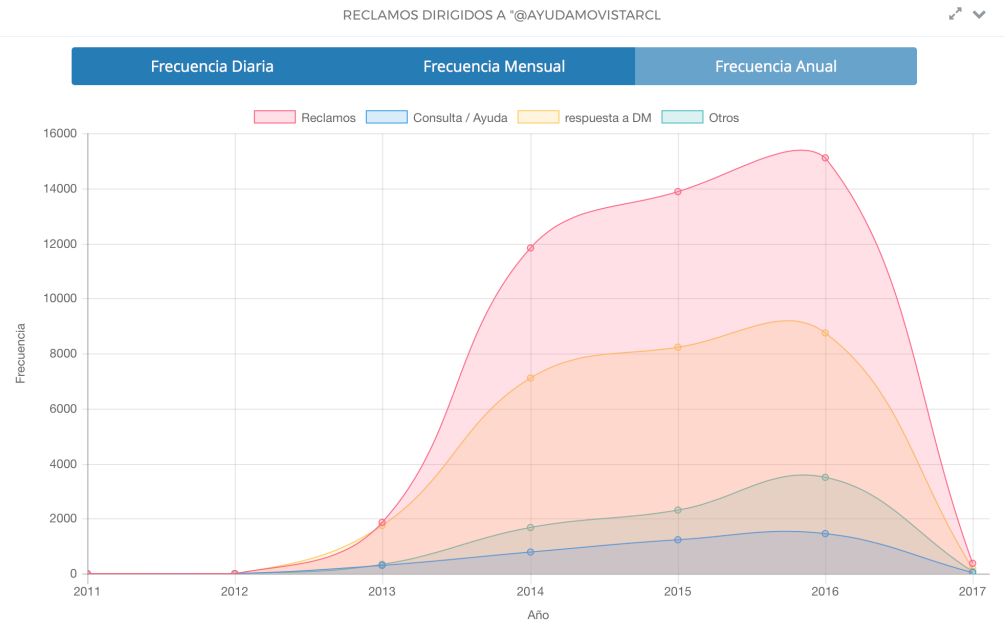

Figura 6.6: Gráfico frecuencia anual - Sección Reclamos
Elaboración propia.

6.3.2. Módulo de Reclamos - Sección Categorías

Esta otra sección del Módulo de Reclamos despliega los datos referentes a la Etapa de Categorización, posee la misma estructura y diseño que la sección Reclamos, permite ver los datos de una cuenta (keyword) específica de Twitter. La figura 6.7 muestra la vista principal de esta sección, la cual parece ser aún más confusa que la anterior, ya que ahora se muestran 9 series de frecuencia, pero también es posible suprimirlas para obtener visualización más simple y particular de alguna categoría.

Al igual que en la otra sección, en esta también es posible desplazarse a través del tiempo con el fin de observar y analizar lo que ha ocurrido con los reclamos en periodos pasados, inclusive hasta la creación de las cuentas en Twitter.

En anexos B.1, B.2 y B.3, se pueden apreciar los gráficos para ocultar series, frecuencia mensual y frecuencia anual respectivamente, que son los mismos mostrados anteriormente, pero ahora para la sección Categorías del Módulo de Reclamos.

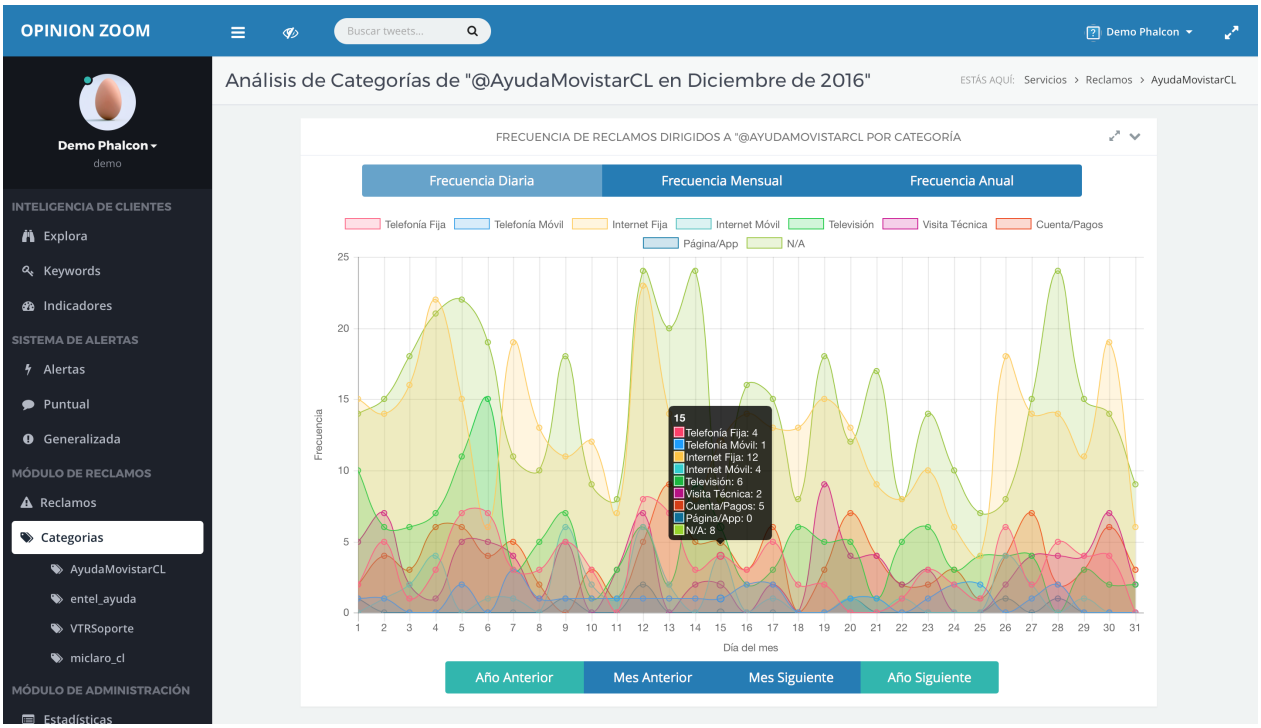

Figura 6.7: Sección Categorías - Módulo de Reclamos OpinionZoom
Elaboración propia.

6.4. Nube de Palabras

OpinionZoom se encuentra en continuo cambio y mejoramiento, es por esto que se realizan iteraciones sobre su modelo de negocio, en donde se entrevistan a potenciales clientes para obtener feedback y de esta manera crear un servicio mejor.

Durante el último proceso de validación del modelo de negocios en Enero de 2016 se visitaron cuatro empresas: Simple Móviles, Netline, Canal cero y Brújula. A ellas se les mostraron los distintos servicios que posee OpinionZoom, dentro de los que se encontraba el Módulo de Reclamos como prototipo funcional.

De estas entrevistas se lograron recibir muchas ideas de como mejorar los servicios de la plataforma y así agregar más valor a las empresas. Dentro de las observaciones recogidas de estas entrevistas en relación al Módulo de Reclamos fue que este les parecía atractivo, se mostraron muy interesados por la información y la funcionalidad que les brindaba, sin embargo, expresaban cierta desconfianza del modelo y el procesamiento de los datos que estaba implícito. Sentían que tan solo mostrar la frecuencia de reclamos o de sus categorías no era suficiente, querían algo más, como por ejemplo visualizar el contenido de los tweets que pertenecían a cada categoría.

Es por esto que se decidió por agregar una nueva funcionalidad al Módulo de Reclamos con el objetivo de solucionar esta inquietud que presentaban los potenciales clientes. En una primera instancia se consideró por mostrar de manera detallada el texto de los tweets perteneciente a cada categoría, sin embargo, esta idea fue descartada dado que los algoritmos desarrollados no poseen una precisión del 100 %, por lo que serían mostrados tweets "mal clasificados", los cuales podrían generar una mala impresión de cara a las empresas al entregar una información errónea.

Como segunda alternativa, la cual fue la escogida finalmente, es generar tópicos por categorías de modo de mostrar los términos más frecuentes en cada una de ellas. Esta solución se compone de dos partes, en una primera es encontrar tópicos distintos dentro de una categoría y luego determinar los términos con mayor frecuencia dentro de estos. La idea de encontrar varios tópicos en una categoría es inspeccionar si es que los comentarios en Twitter hablan de temas diferentes, por ejemplo: en la categoría "Internet fija" pueden existir comentarios relacionados con equipos (modem, router, etc.), conectividad (baja velocidad de navegación, latencia, etc.), entre otros. Por lo que poder determinar si es que existen temas distintos es una herramienta util para las empresas, ya que les entrega aún más información que los términos más comunes en general y la frecuencia de tweets en cada categoría.

El fin principal detrás de esta alternativa es reducir el error del clasificador al mostrar los términos más frecuentes, debido a que la precisión de las 9 clases es superior al 75 %, por lo que encontrar términos muy frecuentes en el otro 25 % (que son los tweets mal clasificados) de los tweets es menos probable ya que son una menor cantidad. Por lo que los términos más ocurrentes que resulten serán referentes a la categoría en cuestión.

Para resolver este problema de encontrar tópicos y términos más frecuentes se utilizó *Topic Model* en su implementación más utilizada: Latent Dirichlet Allocation (LDA). Se utilizó la librería JGibbLDA, la cual se encuentra en el lenguaje de programación JAVA, que es el utilizado en los algoritmos desarrollados durante este trabajo y en la plataforma de OpinionZoom, por lo que es el más adecuado a ocupar.

En primer lugar, para implementar esta nueva funcionalidad del Módulo de Reclamos fue necesario modificar la estructura de la base de datos. Al modelo entidad relación de la figura 6.1 se le agregó una nueva tabla para almacenar los tópicos y términos más frecuentes de cada categoría. Esta nueva tabla se denominó *wordcloud* y se compone de la siguiente estructura:

- **idwordcloud:** Identificador de la tabla wordcloud.
- **word:** Corresponde al término frecuente.
- **topic:** Corresponde al tópico encontrado.
- **categoria:** Representa la categoría a la cual pertenece el tópico y los términos.
- **ano:** Año al cual corresponde el término y tópico.
- **mes:** Mes al cual corresponde el término y tópico.
- **dia:** Día al cual corresponde el término y tópico.
- **weight:** Valor de importancia del término dentro del tópico.
- **keysreclamos_idkeysreclamos:** Llave foránea que identifica la keyword a la cual pertenecen los términos y tópicos.

La figura 6.8 muestra el modelo entidad relación que resulta tras añadir esta nueva tabla antes descrita.

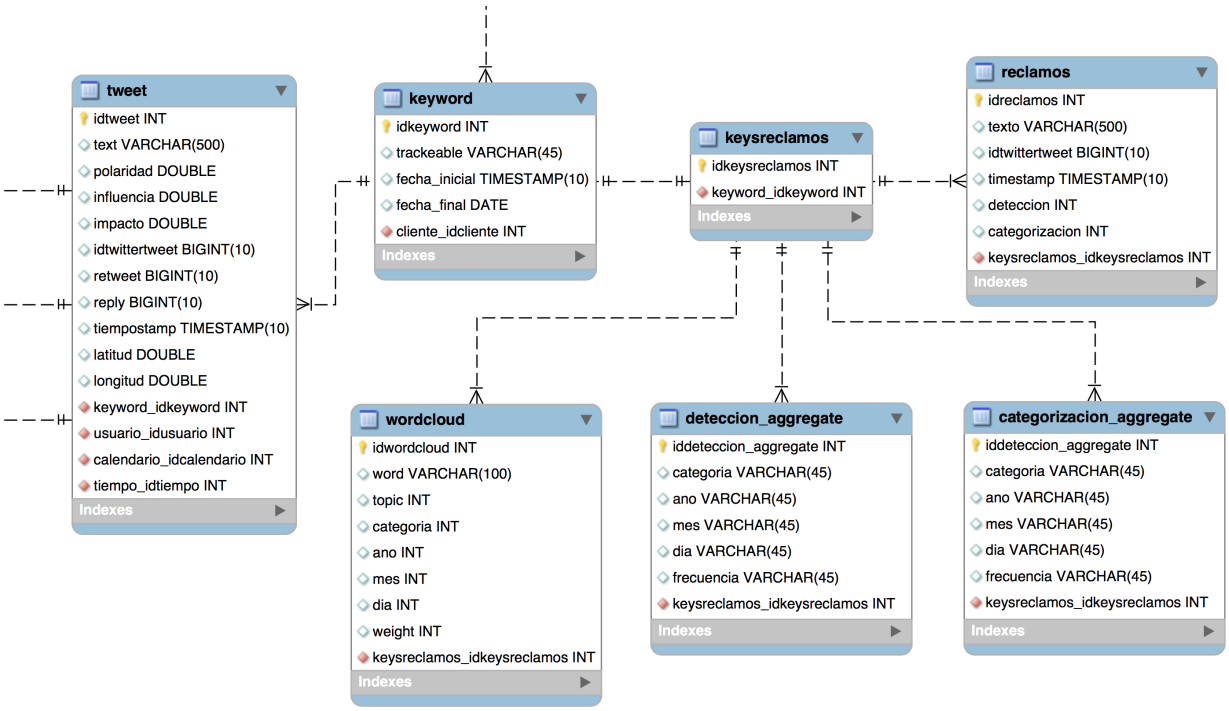

Figura 6.8: Modelo entidad relación final - Módulo de Reclamos
Elaboración propia.

Para aplicar LDA, es necesario definir las características de los tópicos y términos a calcular. Observando la cantidad de datos que llegan a cada una de las cuatro cuentas analizadas se puede concluir que hay una disparidad entre ellas. Existen días en donde estas cuentas de Twitter no reciben tweets de determinadas categorías, por ejemplo en la figura 6.7 se puede apreciar que para el día 15 de Diciembre de 2016, la cuenta @AyudaMovistarCL no recibió ningún mensaje categorizado como *Página / App* (es decir, el modelo no clasificó ningún mensaje en esta categoría). Por otro lado, en ese mismo día se recibieron 12 tweets con relación al servicio de *Internet*, cantidad que parece pequeña para identificar más de un tópico o términos frecuentes que destaquen sobre otros. En base a esto se decidió por abarcar un periodo de tiempo más amplio para calcular los tópicos y términos. Se escogió por considerar un intervalo de una semana como un tiempo adecuado.

Aún considerando este intervalo superior de tiempo, existen categorías que siguen teniendo una baja cantidad de tweets (raras veces nula). Por lo que se decide por condicionar la cantidad de tópicos y términos que se calculen en función de la suma de tweets en el transcurso de la semana en cuestión que se considere. Estas reglas se describen a continuación:

Cantidad de tweets	Número de Tópicos	Número de términos	α	β
Más de 25	3	10	16	0.1
Entre 11 y 25	2	10	25	0.1
Entre 6 y 10	1	5	50	0.1
Menos de 5	1	3	50	0.1

Tabla 6.1: Reglas de Nube de Palabras.
Fuente: Elaboración propia.

Los hiper-parámetros α y β se calculan según lo sugerido en [50], con un valor de $\alpha = 50/K$, donde K corresponde al número de tópicos, y un valor de $\beta = 200/w$ o 0.1.

Para calcular los tópicos y términos más frecuentes de cada categoría de tweets se diseño un algoritmo que implementa la librería JGibbLDA [70], el cual tiene la siguiente secuencia de pasos:

1. **Selección de keywords:** Se seleccionan todas las keywords de la tabla *keysreclamos*.

2. **Extraer tweets:** Para cada keywords se buscan los tweets emitidos durante la última semana. Esta consulta es realizada sobre la tabla *Reclamos* y se realiza una por cada categoría, es decir, 9 consultas distintas.

3. **Preprocesamiento de tweets:** Se realizan técnicas de preprocesado de texto, como eliminar Stopwords, signos de puntuación y otro elementos del texto. Con el objetivo de dejar solo las palabras del tweet.

4. **Almacenamiento intermedio:** Se almacena cada conjunto de tweets en un archivo. La estructura de este archivo lleva en la primera linea el número de tweets y en cada linea siguiente un tweet.

5. **Calcular reglas y parametros:** En base a la cantidad de tweets que contenga cada categoría se determina el número de tópicos, términos e hiper-parámetros en base a la tabla 6.1.

6. **Estimar modelo:** Para cada categoría se estima el modelo correspondiente con la librería JGibbLDA.

7. **Insertar datos:** Para cada modelo calculado se insertan los datos en la tabla *wordcloud*.

Este proceso se repite para todas las keywords de la tabla *reclamos*, que para este trabajo son tan solo cuatro.

La Nube de Palabras tiene el objetivo de mostrar los distintos temas que tratan las categorías, y de esta forma no mostrar los tweets individuales que fueron clasificados, eliminando así el error del modelo de predicción construido. Por esto se decidió por incorporarlo en la sección “Categorías”

y no crear una nueva para exhibir estos datos, de modo de poder visualizar en una misma vista la frecuencia de tweets y los distintos tópicos y términos que emplea la gente para expresarse en una categoría. Para la visualización de los términos más importantes en cada tópico se utilizó la librería "wordcloud2.js" [71].

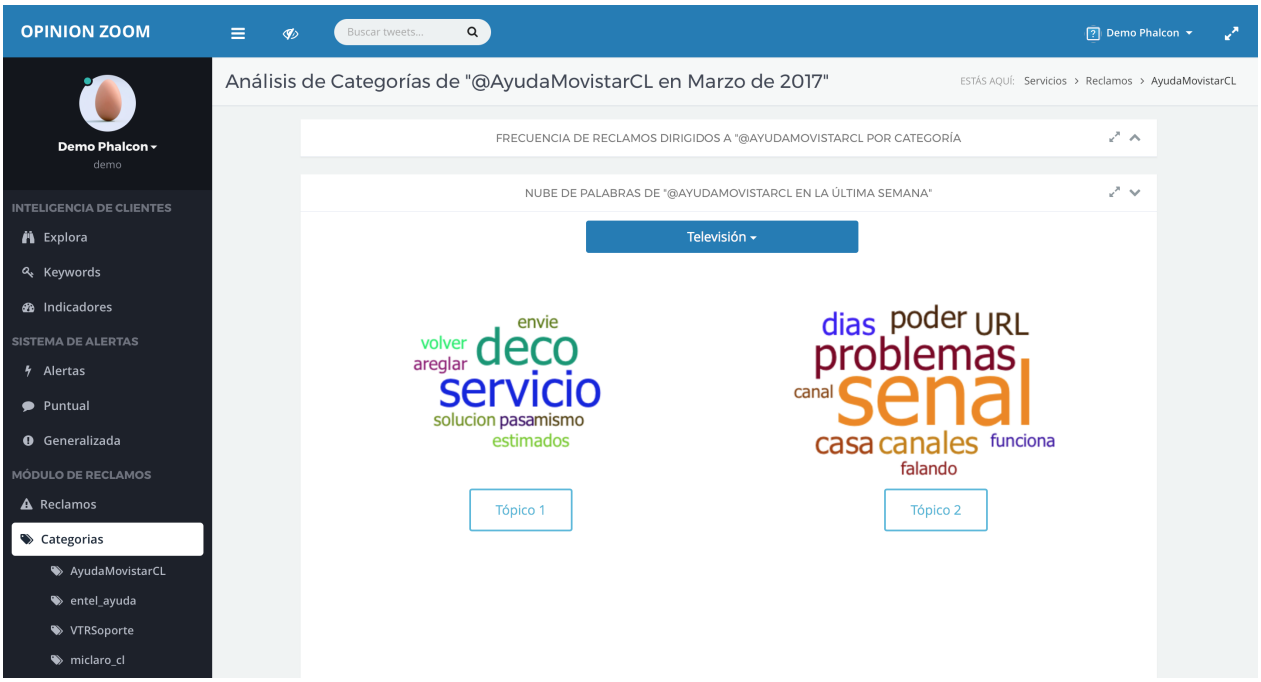

Figura 6.9: Nube de Palabras - Sección Categorías
Elaboración propia.

En la figura 6.9 se muestra la "Sección Categorías" del Módulo de Reclamos, donde ahora consta de la nube de palabras en la parte inferior del gráfico de frecuencia, que para este caso se encuentra minimizado. Se diseñó un menú desplegable en la parte superior del contenedor de la nube con el objetivo de poder observar cada categoría de forma independiente y que no sea confusa su interpretación. En el ejemplo se muestran tan solo 2 tópicos correspondientes a la clase *Televisión*, por lo que durante esa semana se deduce que hubo entre 11 y 25 tweets para dicha categoría.

6.5. Validación con clientes

Tal como se mencionó en un comienzo, el proyecto OpinionZoom está en continuo cambio. Actualmente se encuentra en una segunda iteración del modelo de negocios en donde se están entrevistando a potenciales clientes para obtener feedback de ellos y poder redefinir los servicios y así entregar una plataforma que se adecúe completamente a sus necesidades.

El principal beneficio o utilidad que los clientes encontraron del modulo de reclamos es la frecuencia en los comentarios identificados como reclamos, mostraron interés por conocer cuando existían alzas en la cantidad de reclamos para poder investigar las causas posibles de los eventos, para lo cual la nube de palabras, que fue incorporada en una segunda etapa, les aportaba gran

información, ya que les permitía contextualizar el escenario y el fenómeno desde donde provenían los reclamos. De esta forma no es necesario el mostrar los mensajes a nivel individual, ya que con la información agregada de la nube de palabras los nuevos potenciales clientes entrevistados quedaron satisfechos.

Dentro de los otros intereses mostrados por los potenciales clientes se encuentra el analizar las respuestas de la propia empresa[2] hacia sus clientes, en términos del tiempo que demoran, la cantidad de respuestas (porcentaje de tweets respondidos), etc. Esto debido a que varias empresas tienen estrategias y protocolos establecidos (KPIs) para gestionar las redes sociales, por lo cual disponer de esta información de manera automática les ayuda a llevar una mejor gestión y seguimiento de lo que ocurre a nivel de la red social.

Esta necesidad antes descrita se escapa del alcance de este trabajo por lo que cual se deja propuesta para trabajos futuros e investigaciones del proyecto OpinionZoom.

[2]respuetas a mensajes directos que los usuarios realizan. Estas consultas son respondidas por un área encargada de manejar las redes sociales de las empresas

Capítulo 7

Conclusiones

El presente capítulo da a conocer las conclusiones de este trabajo de título, para posteriormente proponer recomendaciones y trabajos futuros que pueden ser desarrollados.

7.1. Conclusiones

El presente trabajo de título forma parte de las investigaciones del proyecto OpinionZoom, en que se analizan las opiniones depositadas por la comunidad de Twitter, facilitando y mejorando el conocimiento de los clientes para las empresas proveedoras de productos y/o servicios.

Este trabajo presenta la creación de un módulo capaz de extraer tweets, preprocesarlos, identificar si corresponden a reclamos, asignarles una o más categoría y finalmente visualizar los resultados en la plataforma web de OpinionZoom. Específicamente, los resultados se basan en la implementación de un modelo Bag-Of-Words para la clasificación del contenido de Twitter, utilizando como punto inicial lo realizado en [10], en conjunto con las mejores prácticas de [55, 57, 59] para la categorización temática de opiniones en idioma español para Twitter.

El objetivo general de este trabajo es diseñar e integrar un módulo para detectar y categorizar opiniones de reclamos, lo que genera un beneficio para el proyecto OpinionZoom. Se agregó un nuevo servicio a la plataforma, que por lo visto en las entrevistas de la primera iteración del modelo de negocios y en [7] presenta un alto interés por parte de las empresas.

La hipótesis de investigación de este trabajo planteaba el desafío de utilizar algoritmos de Data Mining y Machine Learning para identificar y categorizar opiniones de reclamo en Twitter. Los resultados obtenidos validan la hipótesis, de manera que efectivamente es posible detectar las opiniones de reclamos utilizando solo el texto que los usuarios exponen hacia las empresas en la red social de Twitter.

En cuanto a los objetivos específicos planteados para el desarrollo de este trabajo se explica a continuación el cumplimiento de cada uno de ellos:

- **Evaluar el estado del arte referente a *Topic Classification***: Se realizó una exhaustiva investigación de las técnicas existentes en el preprocesamiento de datos provenientes de Twitter y del estado del arte de la investigación de *Topic Classification* en idioma español para contenido generado en esta red social. En el capítulo 3 se presentan las principales conclusiones de las últimas investigaciones en este ámbito y se exponen las mejores prácticas a realizar.
- **Construir un set de datos de tweets etiquetados que sirva como Data de entrenamiento**: Se optó por desarrollar un modelo clasificador particular por rubro por las diferencias tanto en sus composición como en la forma que los usuarios se expresan. En el capítulo 4 se analiza y se escoge el mejor rubro para verificar la hipótesis planteada. Además, se construyó un juego de datos que abarcó el historial completo de las empresas en Twitter para minimizar sesgos e irregularidades de los datos. Finalmente se etiquetaron los datos y se definieron dos juegos de entrenamiento para diseñar un algoritmo que identificara opiniones de reclamos y otro que las categorizara.
- **Evaluar Algoritmos de clasificación que permitan identificar opiniones de reclamos, y clasificarlas en categorías predefinidas**: En cuanto a los algoritmos, se lograron evaluar 4 distintos en variadas combinaciones de preprocesamiento de texto. En el capítulo 5 se diseñan y escogen los mejores algoritmos para cada uno de los problemas abordados.
- **Diseñar e integrar el módulo funcional en la plataforma web de OpinionZoom**: Finalmente, en el capítulo 6 se implementó de forma exitosa y se validó el módulo funcional, en una primera iteración del modelo de negocios de OpinionZoom, con potenciales clientes.

Con estos 4 objetivos específicos logró cumplir a cabalidad con el objetivo general propuesto. Se diseñó e implemento de forma integra el módulo para detectar y categorizar opiniones de reclamos en OpinionZoom.

Para los resultados obtenidos en la Etapa de Detección de reclamos no existe otra investigación que realice este desafió para realizar una comparación directa y analizar si lo realizado es mejor o peor, lo más cercano fue lo desarrollado en [10], donde se buscó identificar texto con carácter de reclamo en una empresa aseguradora de Alemania. Comparando con dicha investigación, los resultados de este trabajo son más que satisfactorios, se logró un F-Measure general de 0.780 y 0.823 en la clase reclamo, similares a lo que obtuvieron ellos (entre 0.75 y 0.86). Sin embargo, el algoritmo desarrollado de este trabajo tiene la capacidad de clasificar en cuatro clases distintas y no solo dos. Además, lo realizado se encuentra en el dominio de Twitter, donde existen mayores retos que trabajar con texto de una longitud superior.

Analizando en detalle los resultados obtenidos en la Etapa de Categorización, se aprecian ciertas particularidades con respecto a las clases en las cuales los tweets son categorizados. Se obtuvo una *Precision* de 75 % o más para todas clases, sin embargo, por el lado del *Recall* los resultados presentan una mayor variación, la clase "Página web / App" es la que posee el peor rendimiento con un valor de 37.5 %, que se debe principalmente a la baja cantidad de muestras. A modo general, todas las clases presentan un *F-Measure* superior al 50 %.

Realizando una comparación de estos resultados obtenidos en la Etapa de Categorización, se consiguió un desempeño superior a [57, 59]. El rendimiento final del modelo de clasificación construido entregó una *Precision* promedio de 87.8 %, un *F-Measure* de 0.838 y un *Accuracy* de 80.2 %.

El mayor inconveniente de lo realizado en este trabajo de título es que el dominio de aplica-

bilidad de los resultados y modelos obtenidos es muy acotado, tan solo abarca a cuatro empresas pertenecientes al rubro de las telecomunicaciones de Chile. Esta implementación y los resultados que están detrás reflejan el nivel de especificidad de lo realizado, de forma que si se quiere replicar en otro rubro es necesario construir un nuevo juego de datos específico y a partir de este diseñar un algoritmo para cumplir con el desafío de identificar opiniones con carácter de reclamo y clasificar dichos tweets en categorías predefinidas.

Finalmente, el modelo desarrollado se probó en el dominio de *micropagos*[1] en twitter, entregando una *precision* del 70 % y un *recall* del 30 %, por lo que se puede concluir que para ampliar el alcance en la detección de reclamos es necesario desarrollar un set de entrenamiento particular para cada caso, es decir, se debe seguir la misma metodología llevada a cabo en este trabajo de título para la construcción de un clasificador que alcance un desempeño similar al obtenido en otros rubros.

7.2. Trabajo futuro

A partir de lo desarrollado, los resultados obtenidos y los cambios implementados por los comentarios de las empresas entrevistadas, se proponen las siguientes mejoras para el Módulo de Reclamos y los algoritmos de procesamiento que lo conforman:

1. **Evaluar clasificador en otras empresas del rubro:** Se propone analizar y evaluar el comportamiento de los clasificadores construidos en las otras empresas del rubro de las telecomunicaciones que no fueron incluidas en el set de datos de este trabajo. Con respecto a la *Etapa de Detección* no habrían muchos cambios a realizar, ya que todas las empresas en Twitter presentan los mismos tipos de tweets, sin embargo, en la *Etapa de Categorización* es necesario modificar el algoritmo clasificador de modo de adaptarlo a las categorías (servicios) que tenga la empresa a analizar. Por ejemplo para la empresa WOM deberían quitarse las clases que hacen alusión a "Televisión", "Telefonía Fija" e "Internet", ya que no entrega dichos servicios. Esta modificación no tiene mayor complejidad debido a que se diseño un algoritmo de clasificación particular para cada clase en esta segunda etapa, por lo que tan solo se deben suprimir esos algoritmos para aplicarlos en estas otras empresas que no poseen todos los servicios del rubro de las telecomunicaciones.

2. **Evaluar otros algoritmos de Machine Learning:** Los algoritmos evaluados en este trabajo de título no son todos los que existen, varios de estos fueron descartados por límites de procesamiento del hardware utilizado, como lo son Redes neuronales y regresión logística. Se propone analizar el desempeño de estos algoritmos con el fin de verificar si poseen un rendimiento superior al encontrado con los vistos. Para lograr esto es posible realizar reducción de dimensionalidad del problema, análisis de componentes principales, eliminar variables redundantes, variables que posean mucha correlación con otras, etc.

3. **Variar y optimizar parámetros de los algoritmos:** También relacionado con los algoritmos de Machine Learning evaluados, se propone variar y optimizar los parámetros de cada uno de

[1] se utilizó un set de 400 tweets etiquetados manualmente correspondientes a temás de micropagos en 6 temas diferentes: Transporte, comida, aplicaciones móviles, cine, estacionamientos y eventos

ellos, con el fin de lograr mejoras en los resultados obtenidos y así construir un clasificador más preciso. Sin embargo, existe una inconveniente con respecto a este procedimiento, el realizar optimización particular de cada configuración las hace no comparables de una forma directa, ya que no representan el mismo algoritmo.

4. **Inclusión de variables adicionales al texto:** Como lo visto en [55], la inclusión de variables adicionales al texto mejora considerablemente la clasificación temática de tweets. En específico, con la sola incorporación de una variable que segmente a los usuarios se puede mejorar bastante los resultados (En dicha investigación se tuvo un incremento de 18.3 % sobre el modelo de Bag-Of-Words).

5. **Polaridad 2.0:** Tal como se comentó, OpinionZoom se encuentra mejorando sus algoritmos, dentro de los cuales se encuentra el cálculo de polaridad en las opiniones de Twitter, dado esto se propone evaluar la inclusión de esta variable nuevamente en la primera etapa de detección de reclamo, pero esta vez con un clasificador que se desempeñe mejor que el utilizado en el desarrollo de este trabajo, ya que el resultado obtenido sobre si la polaridad del tweet entrega información adicional para determinar si una opinión representa un reclamo no fue positivo, y pudo tener como causa el mal desempeño del clasificador de polaridad que actualmente utiliza OpinionZoom (PAPI).

6. **Tiempo de visualización de datos en el módulo de reclamos:** En el Módulo de Reclamos implementado en OpinionZoom se puede modificar el algoritmo que agrupa los tweets. Se puede disminuir el intervalo de tiempo en que este se realiza, el cual se configuró 1 vez al día, de modo de poder ver en tiempo real la evolución de los reclamos y sus categorías.

7. **Enviar datos clasificados automáticamente:** Por el lado de aplicabilidad de los datos categorizados obtenidos del Módulo de Reclamos, se puede programar y hacer que cada vez que se clasifica un tweet este es enviado (por correo electrónico u otro medio) al área encargada de la categoría en cuestión. Esto con el objetivo de derivar internamente con el área encargada y no dejar que el cliente tenga que comunicarse por otro medio.

8. **Actualización de los datos del clasificador:** Ligado a esta última propuesta, si esto se lleva cabo se puede permitir que el área encargada categorice los tweets que recibe de una mejor forma, y de este modo construir un nuevo set de datos etiquetados por "expertos". Y a partir de este nuevo juego de datos es posible construir un nuevo clasificador mucho más preciso y más actualizado que el desarrollado en este trabajo de título.

9. **Clasificar conversaciones completas:** Tomar todo el contenido de las conversaciones para aplicar un algoritmo que identifique y categorice las opiniones. Lo realizado en este trabajo de título tan solo consideró el primer mensaje que reciben las cuentas en Twitter, con el objetivo de identificar en un comienzo la naturaleza del tweet. Se propone incorporar los otros mensajes que emite el cliente en las conversación para poder clasificar de mejor forma la opinión, ya que como se mencionó en reiteradas oportunidades, una de las mayores dificultades de trabajar con texto proveniente de Twitter es la falta de longitud y contexto de los mensajes, por lo que la inclusión de esta mejora podría afectar de manera positiva los resultados obtenidos.

10. **Evaluar curva de aprendizaje:** El tamaño del set de entrenamiento utilizado en lo desarro-

llado en este trabajo fue calculado como una muestra representativa del universo de tweets, sin embargo, esta cantidad puede no ser la óptima para alcanzar el mejor desempeño de un clasificador. El rendimiento de un algoritmo clasificador mejora cuando se aumenta la cantidad de muestras de las que aprende [72, 73]. Se propone entonces evaluar el desempeño de los algoritmos utilizando una curva de aprendizaje [74], para analizar cual es el número óptimo de muestras a considerar en el set de datos, y de este modo mejorar los resultados obtenidos.

11. **Otros intereses de empresas:** Se plantea como trabajo futuro realizar los 4 intereses propuestos en [7] que no fueron abarcados en este trabajo, estos son:

    - **Tiempo de respuesta:** Se propone analizar la evaluación en el tiempo de las conversaciones que se llevan a cabo en el marco de los reclamos. Considera almacenar los tiempos en que la empresa y el usuario responde a los mensajes, de modo de ver y analizar el comportamiento entre que se genera y se soluciona un reclamo.
    - **Datos de usuario:** Almacenar datos del usuario quien emite los mensaje con el fin de analizar su historial con la empresa. Visualizar mensajes reiterados, reclamos solucionados, reclamos inconclusos, etc.
    - **Identificar cliente:** Determinar si quien emite el reclamo es realmente un cliente de la empresa analizada de modo de no malgastar recursos (para la empresa) contestando reclamos de quienes no son verdaderos clientes. También estos puede ser visto como priorizar los reclamos y comentarios que provienen de clientes reales. Esta propuesta tiene la dificultad de realizar el cruce de información de las bases de datos de la empresa con los datos suministrados por la red social Twitter. Se debe evaluar si es posible realizar este cruce de información de identificar a un usuario de Twitter con una persona real, considerando las implicancias legales.
    - **Otras fuentes de opinión:** Incorporar otras fuentes de opinión y redes sociales donde los usuarios y clientes exponen reclamos, tales como: Facebook, SERNAC, paginas Web de reclamos, etc.

Bibliografía

[1] *World Internet Users Statistics and 2017 World Population Stats*. dirección: http://www.internetworldstats.com/stats.htm (visitado 21 de mar. de 2017).

[2] Subtel, "Resultados Encuesta Nacional de Acceso y Usos de Internet", es, Subsecretaria de Teleomunicaciones, Chile, inf. téc., oct. de 2015, pág. 18. dirección: http://www.subtel.gob.cl/wp-content/uploads/2015/04/Presentacion_Final_Sexta_Encuesta_vers_16102015.pdf (visitado 22 de mar. de 2017).

[3] B. Chile, *Facebook y YouTube se alzan como las redes sociales más usadas por los chilenos*, 2016. dirección: http://www.biobiochile.cl/noticias/tecnologia/moviles/2016/07/01/facebook-y-youtube-se-alzan-como-las-redes-sociales-mas-usadas-por-los-chilenos.shtml (visitado 22 de mar. de 2017).

[4] *Jelly Inmersión*. dirección: http://jelly.cl/inmersion/ (visitado 8 de oct. de 2016).

[5] *Web Intelligence Centre – Sitio Web del WIC del Departamento de Ingenieria Industrial de la Universidad de Chile*. dirección: http://wic.uchile.cl/ (visitado 8 de oct. de 2016).

[6] *OpinionZoom | Bienvenido*. dirección: http://www.opinionzoom.cl/ (visitado 8 de oct. de 2016).

[7] P. d. L. Pollman y F. José, "Uso de la ingeniería de negocios en diseño e implementación de negocio para Start up basada en Web Opinion Mining", es, *Repositorio Académico - Universidad de Chile*, 2015. dirección: http://repositorio.uchile.cl/handle/2250/137850 (visitado 7 de oct. de 2016).

[8] A. Cordova, *Proyecto OpinionZoom: Definición, Documentación y Proyecciones*, es, 2017.

[9] C. Galleguillos y A. Andrés, "Diseño y construcción de un sistema web de análisis de opiniones en twitter integrando algoritmos de data mining", 2015.

[10] S. Ebert y B. Adrian, *Detecting Documents with Complaint Character - Semantic Scholar*, 2013. dirección: /paper/Detecting-Documents-with-Complaint-Character-Ebert-Adrian/df21784524495b3df56528721148818eb3a03ada (visitado 8 de oct. de 2016).

[11] S. B. Kotsiantis, "Supervised Machine Learning: A Review of Classification Techniques", en *Proceedings of the 2007 Conference on Emerging Artificial Intelligence Applications in Computer Engineering: REAL Word AI Systems with Applications in eHealth, HCI, Information Retrieval and Pervasive Technologies*, Amsterdam, The Netherlands, The Netherlands: IOS Press, 2007, págs. 3-24, ISBN: 978-1-58603-780-2. dirección: http://dl.acm.org/citation.cfm?id=1566770.1566773 (visitado 7 de dic. de 2016).

[12] T. O'Reilly, *What Is Web 2.0? Design Patterns and Business Models for the Next Generation of Software*, sep. de 2005.

[13] E. Dans, "La empresa y la web 2.0", *Harvard Deusto marketing & ventas*, vol. 80, págs. 36-43, 2007.

[14] E. Constantinides y S. J. Fountain, "Web 2.0: Conceptual foundations and marketing issues", en, *Journal of Direct, Data and Digital Marketing Practice*, vol. 9, n.º 3, págs. 231-244, ene. de 2008, ISSN: 1746-0166, 1746-0174. DOI: 10.1057/palgrave.dddmp.4350098. dirección: http://link.springer.com/10.1057/palgrave.dddmp.4350098 (visitado 1 de nov. de 2016).

[15] Twitter. (1 de nov. de 2016). Hitos | about, Twitter About, dirección: https://about.twitter.com/es/company/press/milestones (visitado 1 de nov. de 2016).

[16] ——, (1 de nov. de 2016). Company | about, Twitter About, dirección: https://about.twitter.com/company (visitado 1 de nov. de 2016).

[17] ——, (1 de nov. de 2016). Cómo empezar con twitter, Centro de Ayuda de Twitter, dirección: https://support.twitter.com/articles/332061# (visitado 1 de nov. de 2016).

[18] ——, (1 de nov. de 2016). ¿qué son las etiquetas (símbolos "#")?, Centro de Ayuda de Twitter, dirección: https://support.twitter.com/articles/247830# (visitado 1 de nov. de 2016).

[19] JSON, *Json*, nov. de 2016. dirección: http://www.json.org/ (visitado 2 de nov. de 2016).

[20] U. Fayyad, G. Piatetsky-Shapiro y P. Smyth, "From Data Mining to Knowledge Discovery in Databases", *AI Magazine*, vol. 17, n.º 3, pág. 37, mar. de 1996, ISSN: 0738-4602. dirección: http://www.aaai.org/ojs/index.php/aimagazine/article/view/1230 (visitado 1 de dic. de 2016).

[21] J. Han, M. Kamber y J. Pei, *Data Mining: CONCEPTS and Techniques*, 3rd. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2011, ISBN: 978-0-12-381479-1.

[22] I. H. Witten, "Text Mining", English, en *The practical handbook of Internet computing*, OCLC: 54931705, Boca Raton [Fla.: Chapman & Hall/CRC, 2005, págs. 14-1, 14-20, ISBN: 978-1-58488-381-4.

[23] G. G. Chowdhury, "Natural language processing", en, *Annual Review of Information Science and Technology*, vol. 37, n.º 1, págs. 51-89, ene. de 2005, ISSN: 00664200. DOI: 10.1002/aris.1440370103. dirección: http://doi.wiley.com/10.1002/aris.1440370103 (visitado 8 de dic. de 2016).

[24] F. N. Patel y N. R. Soni, "Text mining: A Brief survey", *International Journal of Advanced Computer Research*, vol. 2, n.º 6, págs. 243-248, dic. de 2012, ISSN: 2249-7277, 2277-7970. dirección: https://doaj.org (visitado 16 de ene. de 2017).

[25] Snowball, *StopWords List Spanish*. dirección: https://www.google.com/search?q=snowball+stopwords+list+spanish&ie=utf-8&oe=utf-8&client=firefox-b-ab&gfe_rd=cr&ei=TCF9WOPvN9GnxgSs3ajACw (visitado 16 de ene. de 2017).

[26] A. Z. Broder, S. C. Glassman, M. S. Manasse y G. Zweig, "Syntactic clustering of the Web", en, *Computer Networks and ISDN Systems*, vol. 29, n.º 8-13, págs. 1157-1166, sep. de 1997, ISSN: 01697552. DOI: 10.1016/S0169-7552(97)00031-7. dirección: http://linkinghub.elsevier.com/retrieve/pii/S0169755297000317 (visitado 19 de mar. de 2017).

[27] J. Fürnkranz, *A Study Using n-gram Features for Text Categorization*. 1998.

[28] Y. Zhang, R. Jin y Z.-H. Zhou, "Understanding bag-of-words model: A statistical framework", en, *International Journal of Machine Learning and Cybernetics*, vol. 1, n.º 1-4, págs. 43-52, dic. de 2010, ISSN: 1868-8071, 1868-808X. DOI: 10.1007/s13042-010-0001-0. dirección: http://link.springer.com/10.1007/s13042-010-0001-0 (visitado 16 de ene. de 2017).

[29] A. Rajaraman y J. D. Ullman, *Mining of Massive Datasets*. Cambridge: Cambridge University Press, 2011, ISBN: 978-1-139-05845-2. dirección: http://ebooks.cambridge.org/ref/id/CBO9781139058452 (visitado 16 de ene. de 2017).

[30] E. Alpaydin, "Introduction to Machine Learning", *MIT Press*, vol. 2, 2004. dirección: https://mitpress.mit.edu/books/introduction-machine-learning (visitado 6 de dic. de 2016).

[31] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006, ISBN: 978-0-387-31073-2.

[32] S. Bird, E. Klein y E. Loper, *Natural Language Processing with Python: ANALYZING Text with the Natural Language Toolkit*, en. .º'Reilly Media, Inc.", jun. de 2009, Google-Books-ID: KGIbfiiP1i4C, ISBN: 978-0-596-55571-9.

[33] A. M. Kibriya, E. Frank, B. Pfahringer y G. Holmes, "Multinomial Naive Bayes for Text Categorization Revisited", en *AI 2004: ADVANCES in Artificial Intelligence*, D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, G. I. Webb y X. Yu, eds., vol. 3339, Berlin, Heidelberg: Springer Berlin Heidelberg, 2004,

págs. 488-499, ISBN: 978-3-540-24059-4 978-3-540-30549-1. dirección: http://link.springer.com/10.1007/978-3-540-30549-1_43 (visitado 7 de dic. de 2016).

[34] V. N. Vapnik, *The Nature of Statistical Learning Theory*. New York, NY: Springer New York, 2000, ISBN: 978-1-4419-3160-3 978-1-4757-3264-1. dirección: http://link.springer.com/10.1007/978-1-4757-3264-1 (visitado 7 de dic. de 2016).

[35] ——, *Statistical Learning Theory*, English, 1 edition. New York: Wiley-Interscience, sep. de 1998, ISBN: 978-0-471-03003-4.

[36] U. H.-G. Kreßel, "Advances in Kernel Methods", en, B. Schölkopf, C. J. C. Burges y A. J. Smola, eds., Cambridge, MA, USA: MIT Press, 1999, págs. 255-268, ISBN: 978-0-262-19416-7. dirección: http://dl.acm.org/citation.cfm?id=299094.299108 (visitado 7 de dic. de 2016).

[37] C. Neocleous y C. Schizas, "Artificial neural network learning: A comparative review", en *Hellenic Conference on Artificial Intelligence*, Springer Berlin Heidelberg, 2002, págs. 300-313. dirección: http://link.springer.com/chapter/10.1007/3-540-46014-4_27 (visitado 7 de dic. de 2016).

[38] E. Jaynes, "On the rationale of maximum-entropy methods", *Proceedings of the IEEE*, vol. 70, n.º 9, págs. 939-952, 1982, ISSN: 0018-9219. DOI: 10.1109/PROC.1982.12425. dirección: http://ieeexplore.ieee.org/document/1456693/ (visitado 16 de ene. de 2017).

[39] R. Malouf, "A comparison of algorithms for maximum entropy parameter estimation", en, vol. 20, Association for Computational Linguistics, 2002, págs. 1-7. DOI: 10.3115/1118853.1118871. dirección: http://portal.acm.org/citation.cfm?doid=1118853.1118871 (visitado 16 de ene. de 2017).

[40] L. Rokach y O. Maimon, *Data Mining with Decision Trees: THEORY and Applications*, en, ép. Series in Machine Perception and Artificial Intelligence. WORLD SCIENTIFIC, dic. de 2007, vol. 69, ISBN: 978-981-277-171-1 978-981-277-172-8. dirección: http://www.worldscientific.com/worldscibooks/10.1142/6604 (visitado 8 de dic. de 2016).

[41] J. L. Fleiss, B. Levin y M. C. Paik, *Statistical Methods for Rates and Proportions*, en. John Wiley & Sons, jun. de 2013, Google-Books-ID: 9VefO7a8GeAC, ISBN: 978-1-118-62561-3.

[42] R. Kohavi y F. Provost, "Glossary of Terms, Special Issue on Applications of Machine Learning and the Knowledge Discovery Process", *Machine Learning*, vol. 30, n.º 2-3, págs. 271-274, feb. de 1998, ISSN: 0885-6125. dirección: http://dl.acm.org/citation.cfm?id=288808.288815 (visitado 30 de nov. de 2016).

[43] A. J. Viera y J. M. Garrett, "Understanding interobserver agreement: The kappa statistic", ENG, *Family Medicine*, vol. 37, n.º 5, págs. 360-363, mayo de 2005, ISSN: 0742-3225.

[44] J. R. Landis y G. G. Koch, "The Measurement of Observer Agreement for Categorical Data", *Biometrics*, vol. 33, n.º 1, pág. 159, mar. de 1977, ISSN: 0006341X. DOI: 10.2307/2529310. dirección: http://www.jstor.org/stable/2529310?origin=crossref (visitado 14 de nov. de 2016).

[45] R. Kohavi, "A Study of Cross-validation and Bootstrap for Accuracy Estimation and Model Selection", en *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2*, ép. IJCAI'95, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1995, págs. 1137-1143, ISBN: 978-1-55860-363-9. dirección: http://dl.acm.org/citation.cfm?id=1643031.1643047 (visitado 1 de dic. de 2016).

[46] D. M. Blei y J. D. Lafferty, "Topic models", *Text mining: Classification, clustering, and applications*, vol. 10, pág. 71, 2009.

[47] M.-C. Yang y H.-C. Rim, "Identifying interesting twitter contents using topical analysis", *Expert Systems with Applications*, vol. 41, n.º 9, págs. 4330 -4336, 2014, ISSN: 0957-4174. DOI: http://dx.doi.org/10.1016/j.eswa.2013.12.051. dirección: http://www.sciencedirect.com/science/article/pii/S0957417414000141.

[48] D. Blei, A. Ng y M. Jordan, "Latent dirichlet allocation", *Journal of Machine Learning Research*, vol. 3, n.º 4-5, págs. 993-1022, 2003, cited By (since 1996)4800. dirección: http://www.scopus.com/inward/record.url?eid=2-s2.0-0141607824&partnerID=40&md5=505ce8839ae28d1cb56a7ff91bd0ad2d.

[49] F. A. Vera Cid, "Caracterización de perfiles influyentes en Twitter de acuerdo a tópicos de opinión y la generación de contenido interesante", es, Memoria de Título, Universidad de Chile, 2015. dirección: http://repositorio.uchile.cl/handle/2250/134114 (visitado 6 de abr. de 2016).

[50] T. L. Griffiths y M. Steyvers, "Finding scientific topics", *Proceedings of the National Academy of Sciences*, vol. 101, n.º suppl 1, págs. 5228-5235, 2004.

[51] Christo Boshoff, "An experimental study of service recovery options", *International Journal of Service Industry Management*, vol. 8, n.º 2, págs. 110-130, mayo de 1997, ISSN: 0956-4233. DOI: 10.1108/09564239710166245. dirección: http://www.emeraldinsight.com/doi/full/10.1108/09564239710166245 (visitado 21 de sep. de 2016).

[52] C. Fornell y B. Wernerfelt, "Defensive Marketing Strategy by Customer Complaint Management: A Theoretical Analysis", *Journal of Marketing Research*, vol. 24, n.º 4, págs. 337-346, 1987, ISSN: 0022-2437. DOI: 10.2307/3151381. dirección: http://www.jstor.org/stable/3151381 (visitado 20 de sep. de 2016).

[53] ——, "A Model for Customer Complaint Management", *Marketing Science*, vol. 7, n.º 3, págs. 287-298, 1988, ISSN: 0732-2399. dirección: http://www.jstor.org/stable/183718 (visitado 20 de sep. de 2016).

[54] John A. Schibrowsky y Richard S. Lapidus, "Gaining a Competitive Advantage by Analyzing Aggregate Complaints", *Journal of Consumer Marketing*, vol. 11, n.º 1, págs. 15-26, mar. de 1994, ISSN: 0736-3761. DOI: 10.1108/07363769410053664. dirección: http://www.emeraldinsight.com/doi/full/10.1108/07363769410053664 (visitado 20 de sep. de 2016).

[55] B. Sriram, D. Fuhry, E. Demir, H. Ferhatosmanoglu y M. Demirbas, "Short Text Classification in Twitter to Improve Information Filtering", en *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ép. SIGIR '10, New York, NY, USA: ACM, 2010, págs. 841-842, ISBN: 978-1-4503-0153-4. DOI: 10.1145/1835449.1835643. dirección: http://doi.acm.org/10.1145/1835449.1835643 (visitado 21 de oct. de 2016).

[56] *Weka 3 - Data Mining with Open Source Machine Learning Software in Java*. dirección: http://www.cs.waikato.ac.nz/ml/weka/ (visitado 1 de mar. de 2017).

[57] A. Fernández Anta, L. Núñez Chiroque, P. Morere y A. Santos Méndez, "Sentiment analysis and topic detection of Spanish tweets: A comparative study of NLP techniques", eng, mar. de 2013, ISSN: 1135-5948. dirección: http://rua.ua.es/dspace/handle/10045/27863 (visitado 24 de oct. de 2016).

[58] J. Villena-Román, S. Lana-Serrano, E. Martínez-Cámara y J. C. González-Cristóbal, "TASS - Workshop on Sentiment Analysis at SEPLN", es, *Procesamiento del Lenguaje Natural*, vol. 50, n.º 0, págs. 37-44, abr. de 2013, ISSN: 1989-7553. dirección: http://journal.sepln.org/sepln/ojs/ojs/index.php/pln/article/view/4657 (visitado 26 de oct. de 2016).

[59] F. Batista y R. Ribeiro, "Sentiment Analysis and Topic Classification based on Binary Maximum Entropy Classifiers", es, *Procesamiento del Lenguaje Natural*, vol. 50, n.º 0, págs. 77-84, abr. de 2013, ISSN: 1989-7553. dirección: http://journal.sepln.org/sepln/ojs/ojs/index.php/pln/article/view/4662 (visitado 26 de oct. de 2016).

[60] G. Salton y C. Buckley, "Term-weighting approaches in automatic text retrieval", *Information Processing & Management*, vol. 24, n.º 5, págs. 513-523, ene. de 1988, ISSN: 0306-4573. DOI: 10.1016/0306-4573(88)90021-0. dirección: http://www.sciencedirect.com/science/article/pii/0306457388900210 (visitado 26 de oct. de 2016).

[61] S. Departamento de Estudios e Inteligencia Servicio Nacional del Consumidor, "Estudios Descriptivo del E-Commerce en Chile y Análisis de Reclamos ante SERNAC.", es, Ministerio de Economía, Fomento y Turismo., Chile, inf. téc., ago. de 2014. dirección: http://www.sernac.cl/wp-content/uploads/2014/08/Reporte_E_Commerce_Reclamos-2013-2014.pdf (visitado 28 de oct. de 2016).

[62] SERNAC, "Ranking-del-Mercado-Financiero-sub-mercado-Banca.pdf", es, Ministerio de Economía, Fomento y Turismo., Chile, inf. téc., dic. de 2015. dirección: http://www.sernac.cl/wp-content/uploads/2015/12/Ranking-del-Mercado-Financiero-sub-mercado-Banca.pdf (visitado 4 de nov. de 2016).

[63] S. Subtel, "Ranking de reclamos sernac subtel", es, Subtel y SERNAC, Chile, inf. téc., nov. de 2015, pág. 28. dirección: http://www.subtel.gob.cl/wp-content/uploads/2015/12/ranking_reclamos_sernac_subtel_octubre2015.pdf (visitado 7 de nov. de 2016).

[64] C. Sánchez y V. David, "Diseño e implementación de un sistema para monitorear el consumo y opinión sobre la marihuana en Twitter", es, *Repositorio Académico - Universidad de Chile*, 2016. dirección: http://repositorio.uchile.cl/handle/2250/141030 (visitado 23 de nov. de 2016).

[65] V. A. Hernández Martínez, "Identificación de la presencia de ironía en el texto generado por usuarios de Twitter utilizando técnicas de Opinion Mining y Machine Learning", es, 2015. dirección: http://repositorio.uchile.cl/handle/2250/134793 (visitado 6 de abr. de 2016).

[66] E. Marrese Taylor, "Diseño e implementación de una aplicación de web opinion mining para identificar preferencias de usuarios sobre productos turísticos de la X región de Los Lagos", en, *Repositorio Académico - Universidad de Chile*, 2013. dirección: http://repositorio.uchile.cl/handle/2250/113464 (visitado 6 de abr. de 2016).

[67] H. Cordobés, A. Fernández Anta, L. Chiroque, F. Pérez, T. Redondo y A. Santos, "Graph-based Techniques for Topic Classification of Tweets in Spanish", en, *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 2, n.º 5, pág. 31, 2014, ISSN: 1989-1660. DOI: 10.9781/ijimai.2014.254. dirección: http://www.ijimai.org/journal/node/590 (visitado 21 de oct. de 2016).

[68] J.-A. J.-M. Balazs Thenot, "Diseño, desarrollo e implementación de una aplicación de web opinion mining para identificar el sentimiento de usuarios de Twitter con respecto a una compañia de retail", es, Memoria de Título, Universidad de Chile, 2015. dirección: http://repositorio.uchile.cl/handle/2250/137769 (visitado 6 de jun. de 2016).

[69] *Chart.js | Open source HTML5 Charts for your website*. dirección: http://www.chartjs.org/ (visitado 23 de mar. de 2017).

[70] X.-H. Phan y C.-T. Nguyen, *Jgibblda: A Java Implementation of Latent Dirichlet Allocation (LDA) using Gibbs Sampling for Parameter Estimation and Inference*, 2008. dirección: http://jgibblda.sourceforge.net/ (visitado 19 de mar. de 2017).

[71] timdream, *Wordcloud2.js - tag cloud/Wordle presentation on 2d canvas or HTML*, 2016. dirección: https://timdream.org/wordcloud2.js/#love (visitado 23 de mar. de 2017).

[72] K. Nigam, A. K. Mccallum, S. Thrun y T. Mitchell, "Text Classification from Labeled and Unlabeled Documents using EM", en, *Machine Learning*, vol. 39, n.º 2-3, págs. 103-134, ISSN: 0885-6125, 1573-0565. DOI: 10.1023/A:1007692713085. dirección: http://link.springer.com/article/10.1023/A:1007692713085 (visitado 12 de oct. de 2016).

[73] M. Banko y E. Brill, “Mitigating the Paucity-of-data Problem: Exploring the Effect of Training Corpus Size on Classifier Performance for Natural Language Processing”, en *Proceedings of the First International Conference on Human Language Technology Research*, ép. HLT ’01, Stroudsburg, PA, USA: Association for Computational Linguistics, 2001, págs. 1-5. DOI: 10.3115/1072133.1072204. dirección: http://dx.doi.org/10.3115/1072133.1072204 (visitado 12 de oct. de 2016).

[74] R. L. Figueroa, Q. Zeng-Treitler, S. Kandula y L. H. Ngo, “Predicting sample size required for classification performance”, *BMC Medical Informatics and Decision Making*, vol. 12, pág. 8, 2012, ISSN: 1472-6947. DOI: 10.1186/1472-6947-12-8. dirección: http://dx.doi.org/10.1186/1472-6947-12-8 (visitado 12 de oct. de 2016).

Anexo A

Lista de Stopwords

Lista de Stopwords				
a	estaréis	habidos	nuestra	tendríamos
al	estaremos	habiendo	nuestras	tendrían
algo	estaría	habrá	nuestro	tendrías
algunas	estaríais	habrán	nuestros	tened
algunos	estaríamos	habrás	o	tenéis
ante	estarían	habré	os	tenemos
antes	estarías	habréis	otra	tenga
como	estas	habremos	otras	tengáis
con	estás	habría	otro	tengamos
contra	este	habríais	otros	tengan
cual	esté	habríamos	para	tengas
cuando	estéis	habrían	pero	tengo
de	estemos	habrías	poco	tenía
del	estén	han	por	teníais
desde	estés	han	porque	teníamos
donde	esto	has	que	tenían
durante	estos	hasta	qué	tenías
e	estoy	hay	quien	tenida
el	estuve	haya	quienes	tenidas
él	estuviera	hayáis	se	tenido
ella	estuvierais	hayamos	sea	tenidos
ellas	estuviéramos	hayan	sea	teniendo
ellos	estuvieran	hayas	seáis	ti
en	estuvieras	he	seamos	tiene
entre	estuvieron	hemos	sean	tiene
era	estuviese	hube	seas	tienen
erais	estuvieseis	hubiera	ser	tienes
éramos	estuviésemos	hubierais	será	todo
eran	estuviesen	hubiéramos	serán	todos

Continuación de lista de Stopwords				
eras	estuvieses	hubieran	serás	tu
eres	estuvimos	hubieras	seré	tú
es	estuviste	hubieron	seréis	tus
es	estuvisteis	hubiese	seremos	tuve
esa	estuvo	hubieseis	sería	tuviera
esas	fue	hubiésemos	seríais	tuvierais
ese	fue	hubiesen	seríamos	tuviéramos
eso	fuera	hubieses	serían	tuvieran
esos	fuerais	hubimos	serías	tuvieras
esta	fuéramos	hubiste	set	tuvieron
está	fueran	hubisteis	sí	tuviese
está	fueras	hubo	sido	tuvieseis
estaba	fueron	la	siendo	tuviésemos
estaba	fueron	las	sin	tuviesen
estabais	fuese	le	sois	tuvieses
estabamos	fueseis	les	somos	tuvimos
estábamos	fuésemos	lo	son	tuviste
estaban	fuesen	los	son	tuvisteis
estabas	fueses	más	soy	tuvo
estad	fui	me	su	tuya
estada	fuimos	mi	sus	tuyas
estadas	fuiste	mí	suya	tuyo
estado	fuisteis	mía	suyas	tuyos
estado	ha	mías	suyo	un
estados	ha	mío	suyos	una
estados	habéis	míos	también	uno
estáis	haber	mis	tanto	unos
estamos	había	mucho	te	vosotras
están	había	muchos	tendrá	vosotros
están	habíais	muy	tendrán	vuestra
estando	habíamos	nada	tendrás	vuestras
estar	habían	ni	tendré	vuestro
estará	habías	no	tendréis	vuestros
estarán	habida	nos	tendremos	y
estarás	habidas	nosotras	tendría	ya
estaré	habido	nosotros	tendríais	yo

Tabla A.1: Lista de Stopwords - Snowball

Anexo B

Gráficos Módulo de Reclamos - Sección Categorías

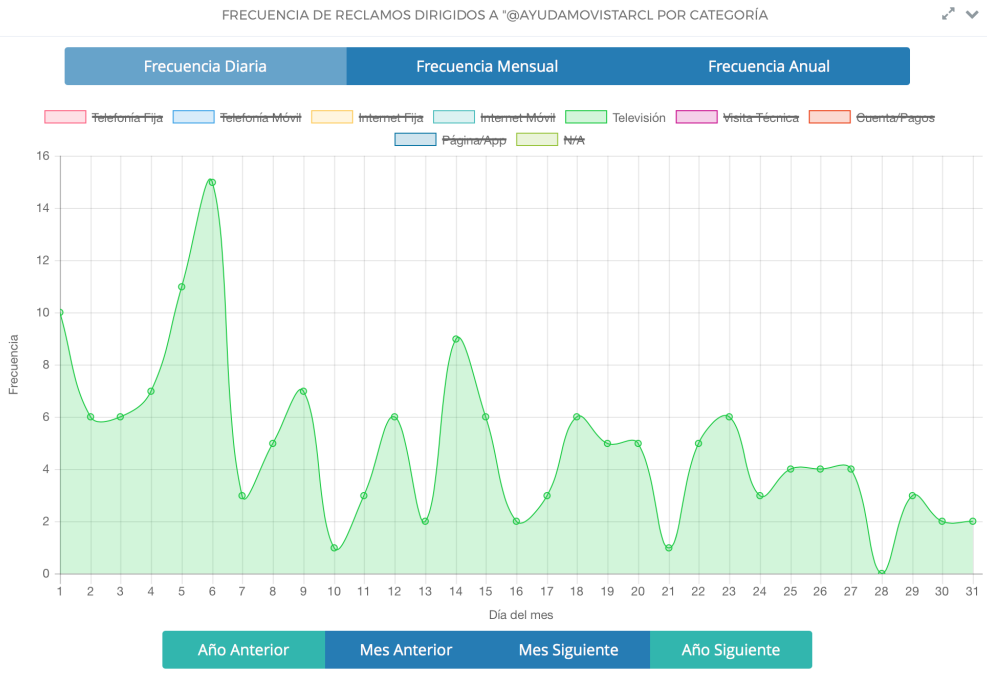

Figura B.1: Gráfico solo tweets “Televisión” - Sección Categorías
Elaboración propia.

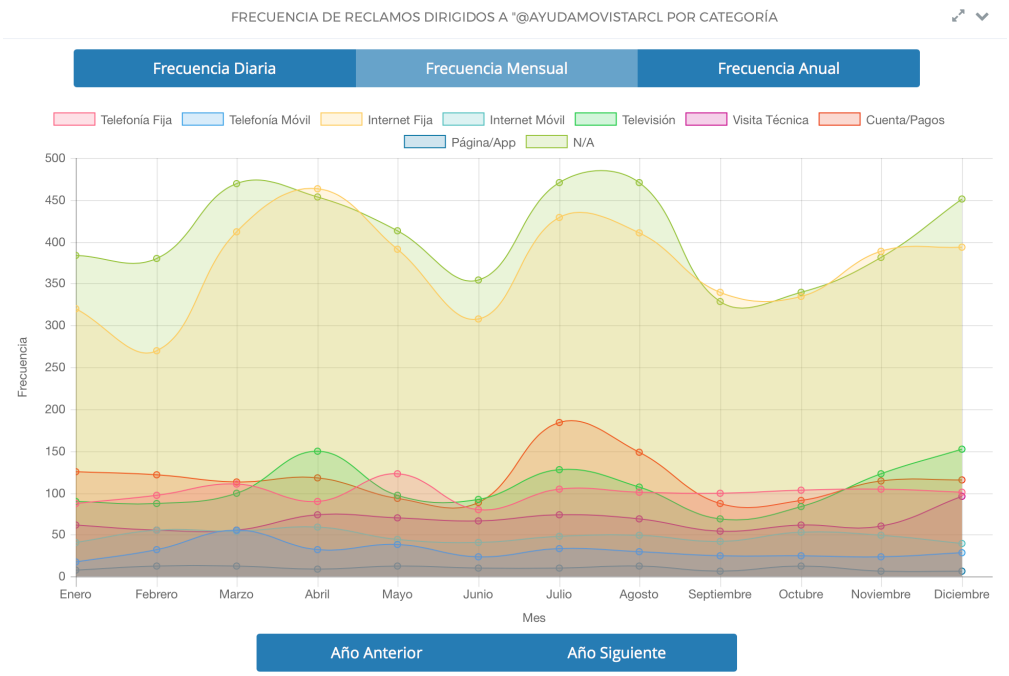

Figura B.2: Gráfico frecuencia mensual - Sección Categorías
Elaboración propia.

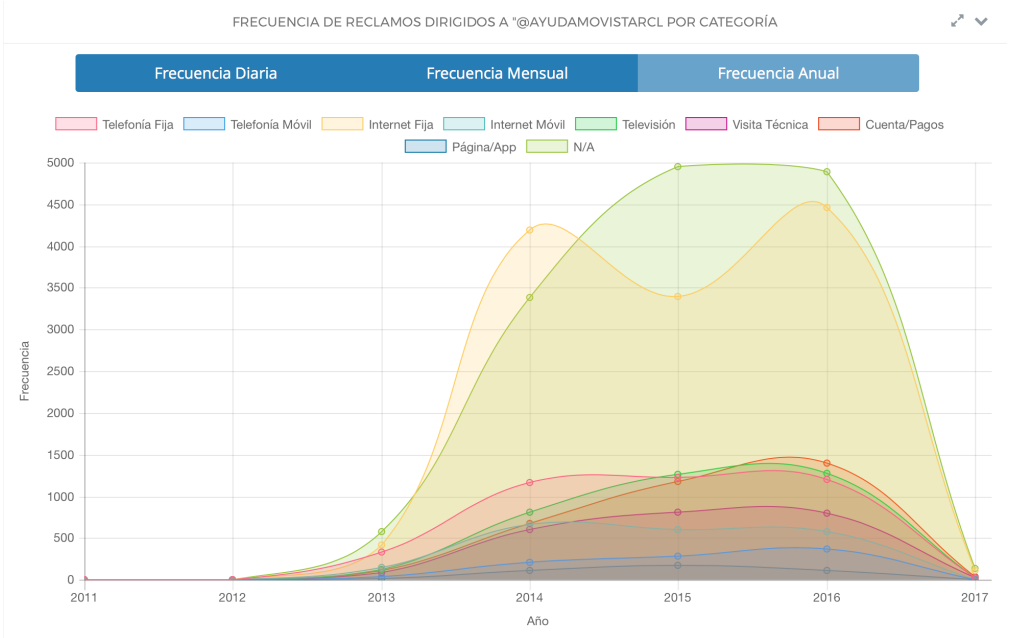

Figura B.3: Gráfico frecuencia mensual - Sección Categorías
Elaboración propia.

Anexo C

Resumen Entrevista Entel

Entrevista realizada el 25 de Septiembre de 2016 a Gonzalo Cortés De La Piedra - Gerente Nacional Ventas Territorios Entel.

C.1. Reclamos en Entel

Entel básicamente vende el servicio de conectividad entre dispositivos móviles, no posee equipos propios, salvo ahora último que tiene su propia marca de teléfonos Öwn hechos en China.

La industria de las telecomunicaciones móviles es una industria muy dinámica en el sentido que ciclos de los productos y servicios son cortos, menos de 5 años. Antiguamente el núcleo del negocio se encontraba en "voz" (minutos), mientras que hoy en día el foco de las empresas son los "datos de navegación" se tuvieron que adaptar tanto las antenas físicas, las personas, ejecutivos y sistemas.

Las antenas se tuvieron que actualizar, pasando por diversas tecnologías: 2g, 3g, 4g y LTE+; lo que trajo una inversión inicial alta, pero con costos de mantención mínimos. Lo que lleva a que después de instalar las antenas hay prácticamente solo ganancias.

Los ejecutivos de venta se tuvieron que capacitar frente a las nuevas tecnologías, y equipos que salieron en el mercado, con el fin de poder entregarle la información a los clientes que acuden presencialmente a las tiendas. Cabe destacar que los chilenos no les gusta esperar, por lo que prefieren comprar los equipos de manera presencial para poder llevárselos en el mismo instante. En este punto Entel está enfocando grandes esfuerzos para cambiar la forma de contacto con el consumidor, tratando de mejorar sus plataformas web de modo que sea más intuitivo y fácil realizar ventas de productos y servicios a través de internet, ya que es más barato que tener a ejecutivos en las tiendas para vender.

Los sistemas también se han tenido que adaptar, agregando "parches" a los softwares existentes cada vez que sale alguna tecnología nueva. El ejemplo más drástico sucedió cuando se cambió el modelo de negocio, de tarifar por minutos a tarifar por datos móviles. Esto trajo como consecuencia

un cambio total en los sistemas (ERP) de la empresa ya que no estaban adaptados para el nuevo modelo de negocios.

En relación a los reclamos que se dan en la empresa Entel se pueden establecer distintas categorías:

1. **Reclamos Adultos 40+ años:** Hay reclamos que provienen principalmente de adultos, mayores a 40 años, que les cuesta entender el dinamismo que tiene esta industria. No comprenden la lógica ente datos móviles y minutos, o no entienden porque los datos son limitados siendo que en los servicios hogar no lo es así. Al momento de contratar algún servicio o comprar de equipos siempre quedan con dudas que evitan preguntar para no quedar como ignorantes, por lo que en el futuro vuelven a consultar o reclamar por temas que están explícitamente en los contratos o que son básicos y que evitaron preguntar en el momento oportuno cuando adquirieron el servicio.

2. **Reclamos de planes:** Otro tipo de reclamo que se da con frecuencia son los referentes a los planes: sus tarifas, minutos, datos móviles, duración, facturación, etc. Principalmente se reciben quejas en relación a que la competencia ofrece los mismo o mejores condiciones por los mismos precios. (el modo de operar de Entel frente a esto se encuentra explicado más adelante)

3. **Reclamos de señal:** Existen además reclamos sobre la señal que entrega la empresa. Siendo que Entel es percibida como la mejor empresa en calidad de servicio en el rubro de la telefonía móvil. Aun así, es imposible "llegar a todos lados", el radio de las antenas es variable en el tiempo, por lo que, si existen una alta demanda en un punto, el alcance de la antena disminuye, quedando usuarios con mal servicios durante la congestión de ese punto. También las antenas están posicionadas son extrema precisión y abarcan distintas por sobre los 10 kms., por lo que existen casos donde aves chocan contra las antenas y modifican el área de las antenas y por ende muchos usuarios quedan sin servicio lo que conlleva a reclamos de este tipo.

4. **Reclamos de equipos:** También hay reclamos con relación a la configuración de equipos, específicamente con la tecnología de transmisión de datos móviles (2g, 3g, 4g, LTE+). Principalmente son reclamos sobre la variación de velocidad de navegación. Para esto Entel cuenta con "puntos Smart" en sus tiendas, donde un ejecutivo especialista en el tema, trata de solucionar estos problemas. En mayor detalle se explica que existen sitios de alta demanda donde se saturan las antenas y por ende el ancho de banda disponible no da abasto, y que en muchos casos la tecnología 3g es más rápida que la 4g debido a la saturación de los puntos.

5. **Reclamos técnico estructural:** Corresponden a los reclamos que realizan los clientes en relación a la cobertura que posee la empresa, el trazado de las antenas no contempla el 100 % del territorio, por lo que existen zonas muertas en donde no hay señal y los clientes reclaman con relación a esto y es necesario explicarles esto con mayor detalle.

6. **Reclamos de contrato:** Existe reclamos se parte de los clientes en relación a los contratos de arriendo de los equipos, desconocen el sistema como se entregan los equipos móviles. Reclaman sobre el estar "atados" a la compañía por 18 meses sin poder cambiarse. Tampoco comprenden muchas veces que el equipo no es de ellos hasta haber pagado todas las cuotas

de arriendo.

7. **Reclamos de satisfacción:** Finalmente, el último tipo de reclamos que recibe la empresa Entel es relación a que los productos y servicios entregados no satisfacen las necesidades o condiciones que los clientes requieren.

En resumen, de todas las categorías identificadas de reclamos de los clientes hacia la empresa se puede destacar que principalmente hacen referencia al desconocimiento que existe por parte de los usuarios frente a los servicios que contratan. Del total de reclamos que se reciben a nivel país, aproximadamente entre un 70 y 80 % corresponden a reclamos asociados al mal uso de los datos móviles. Finalmente, las acciones que realizar la empresa Entel se concentra en generar conciencia del uso de los servicios de manera apropiada a sus clientes.

C.2. Modelo de retención de clientes

Entel cuenta con un modelo de retención de clientes, en casos de reclamos donde el consumidor se haya visto afectado de manera negativa y existan causas verificables. Este modelo presenta 3 niveles:

1. En primer lugar, existe un ejecutivo que se encarga de manejar estos casos de manera presencial en las tiendas, posee protocolos y herramientas (facultades) para realizar esto. Si el caso lo amerita se pueden hacer entregas de descuentos parciales o permanentes en los servicios contratados por los clientes, cambios de equipo antes de tiempo, regalo de puntos, etc. Todas estas acciones con el fin de retener al cliente y evitar que se fugue hacia la competencia.

2. En segundo lugar, existe un "call center" en Santiago, especialista en retención de clientes y que posee estrategias más agresivas para tratar de retener a los clientes que posean algún descontento o reclamos con la empresa. Estas acciones son del mismo estilo que las primeras, pero en mayor grado. Todo esto siempre que sean consecuentes con el caso. No cualquiera puede solicitar beneficios sin una razón de peso.

3. Como último nivel se tiene la renuncia del cliente con todos los pasos y procedimientos que esto conlleva y su fuga de la empresa.

Entel posee un procedimiento identificar problemas en sus nodos o puntos (antenas). Poseen un sistema que les permite trackear todos sus nodos y las áreas geográficas que abarcan, por lo que, al recibir un reclamo de problemas de señal, ubican el punto en el mapa e identifican el nodo a cuál pertenece. El sistema les genera alerta cuando existen varios (umbral) reclamos de señal sobre el mismo nodo. El procedimiento que realizan en consecuencia en primer lugar consiste en un reinicio del nodo y de no ser exitoso deben ir a terreno a inspeccionar y reparar en caso de ser necesario dicho punto (antena). Este sistema genera grandes cantidades de información en tiempo real asociada a los problemas y reparos de cada nodo dentro del sistema, pero esta información es muy técnica para entregársela al cliente, por lo que se decide por no informarle a los clientes que el servicio de señal fue restablecido.

Este sistema permite analizar la información de forma agregada y trackear el estado de la red de

forma global y particular

En base a los protocolos que los ejecutivos de las tiendas y en los servicios de call center (103), estos están siendo capacitados constantemente acerca de las nuevas tecnologías y problemas que puedan suceder. Poseen una "wiki" de los posibles problemas y pasos a seguir para solucionarlos.

Además, en sus tiendas poseen módulos de auto atención en donde los clientes se pueden comunicar con especialista en Santiago de manera telefónica a modo de solucionar sus problemas.

Entel se encuentra enfocando grandes cantidades de recursos en sus sistemas online para hacer sus sistemas más dinámicos y que los clientes puedan interactuar de manera más fácil y rápida. Poseen en su Web información referente a problemas comunes y sus posibles soluciones de modo que los clientes puedan encontrar información fácilmente y no recurrir a otros medios. La justificación de esta iniciativa es el alto costo de mantener las tiendas, sin embargo, existen zonas o comunas donde es necesario mantener las tiendas físicas para mantener la imagen de la empresa (cercanía a la gente)

Vicepresidencias de Entel:

1. Vicepresidencia personas y tiendas

2. Vicepresidencia clientes empresas

3. Vicepresidencia corporaciones (grandes empresas)

4. Vicepresidencia operaciones y redes

Se definen estas 4 áreas distintas en relación a las líneas de negocio que poseen, ya que atienden a clientes distintos y para cada uno poseen productos y servicios diferentes, además de poseer canales de venta y post venta totalmente distintos. La vicepresidencia de operaciones y redes es transversal a las otras 3. Dentro de la proporción de clientes que posee Entel alrededor de un 90-95

C.3. Entel y Twitter

En relación a lo que realiza la empresa Entel en Twitter se destaca en tener una persona encargada de cada una de las cuentas que esta posee, las que están enfocada en base a las líneas de negocio que esta posee:

1. **@entel:** Principalmente para dar a conocer nuevos productos, promociones, concursos a sus clientes, tiene como objetivo ser un medio de marketing.

2. **@entel_ayuda:** Se encarga de recibir los reclamos y las consultas acerca de los productos y servicios que la empresa posee.

3. **@entel_empresas:** Corresponde a la cuenta de sus servicios enfocados a medianas y pequeñas empresas.

4. **@zonaEntel:** Es la cuenta encargada de comunicar noticias sobre el club de beneficios de Entel.

La cuenta @entel_ayuda se encarga de recepcionar los reclamos y consultas que los clientes y potenciales clientes exponen en la red social de Twitter, para esto cuentan con un protocolo que les indica que responder en cada caso y de redirigir cada caso al área correspondiente, por lo que les interesa identificar qué tipo de mensaje la gente publica en Twitter y dentro de que categoría de reclamos cae.

En otras palabras, les interesa diferenciar si una opinión corresponde a un reclamo, consulta, opiniones despectivas, etc. Además de clasificar los reclamos dentro de sus categorías de reclamos para realizar la gestión de manera más rápida y automatizada.

También les interesa el identificar la posición geográfica desde donde se emiten los reclamos, en el caso de los reclamos de señal les interesa cruzar los datos con su sistema de trackeo de nodos para identificar deficiencias en los servicios.