Disertación

Maestría en Ingeniería Informática – Computación Movil

Mobile Application for Analysis of Sentiments in Twitter

Juan Pablo Guevara Gordillo

Leiria, June of year 2018

Dissertation

Master in Informatic Engineering – Mobile Computing

Mobile Application for Analysis of Sentiments in Twitter

Juan Pablo Guevara Gordillo

Dissertation developed under the supervision of professor Catarina Helena Branco Simões da Silva, professor at the School of Technology and Management of the Polytechnic Institute of Leiria and co-supervision of professor Joana Madeira Martins Costa *and by* Mario Raúl Morales Morales, professor at the Universidad Central del Ecuador.

Leiria, *July 23 of year 2018*

Dedicatoria

La presente tesis se la dedico principal y especialmente a mis padres y abuelos maternos, ya que gracias a su apoyo, paciencia y confianza he culminado mi carrera, gracias por estar siempre a mi lado apoyándome y aconsejándome incondicionalmente.

A mis hermanos por estar siempre presentes orientándome en cada una de las decisiones importantes de mi vida.

Además, debo agradecer a todos aquellos excelentes Docentes que, con sus conocimientos, sabios consejos y apoyo continuo, contribuyeron no solo en darme una formación académica respetable sino también enriquecieron mi formación personal, así como el apoyo sincero e incondicional de aquellos buenos amigos y compañeros que siempre estuvieron a mi lado.

Agradecimientos

A Dios por sus múltiples bendiciones y permitirme cumplir mis objetivos.

A mis Padres César Guevara y Martha Gordillo por ser mi motivación, mis amigos, mis confidentes y mí guía.

A mis hermanos Gabriel y Leonardo, por brindarme sus consejos, apoyo y ánimo.

A todos mis profesores del Instituto Politécnico de Leiria en especial:

A mis Tutores de Tesis Dra. Catarina Silva, Dra. Joana Costa, Msc. Mario Morales y Msc. Jorge Arroba quien, con su esfuerzo y dedicación, supieron guiarme en el desarrollo de este proyecto de tesis.

A todas las personas que forman parte de mi vida que de una u otra forma me han brindado su valiosa amistad y colaboración.

A mis tíos: Jaime, Mauricio, Cristina y mis abuelos maternos quienes, con su profesionalismo, experiencia y conocimientos fue un gran aporte para el desarrollo de este proyecto de tesis.

Al SENESCYT que me ha brindado la oportunidad y su confianza para seguir mis estudios en esta prestigiosa institución.

A la Universidad Central del Ecuador y al Instituto Politécnico de Leiria por ayudarme durante el proceso académico.

Nota Previa

Del trabajo efectuado resulto las siguientes publicaciones:

- Juan Guevara, Joana Costa, Jorge Arroba, Catarina Silva. “Harvesting Opinions in Twitter for Sentiment Analysis “. CISTI'2018 - 13ª Conferencia Ibérica de Sistemas y Tecnologías de Información. Juan Guevara realizo la ponencia del trabajo. Publicada en el 2018.
- Juan Guevara, Joana Costa, Jorge Arroba, Catarina Silva,” Análisis de Sentimientos en Twitter para alertas tempranas de femicidios”. I Jornadas Temáticas en Sistemas de Información e Inteligencia Artificial (Poster). Febrero 2018.

Resumen

El Análisis de Sentimientos es una técnica muy popular para el estudio de redes sociales. Una de las redes sociales más populares para microblogging, con gran crecimiento, es Twitter, ya que permite a las personas expresar sus opiniones utilizando oraciones cortas y simples. Estos textos se generan a diario y por esta razón, es común que las personas quieran saber cuáles son los temas de actualidad y sus derivaciones. En este trabajo, proponemos implementar una aplicación móvil que brinde información a las personas, como un grado de polaridad positiva o negativa, sobre cualquier tema relevante en la sociedad, ayudando de esta manera a que las personas puedan tomar la mejor decisión. En el aplicativo se utilizarán varias técnicas de clasificación de texto de manera conjunta. Estas técnicas están enfocadas en el aprendizaje de máquina y de léxico.

Palabras clave: Análisis de Sentimientos, Twitter, Léxico, Aprendizaje de Máquina.

Abstract

Sentiment analysis is a very popular technique for the study of social networks. One of the most popular social networks for microblogging that has a great growth is Twitter, which allows people to express their opinions using short and simple sentences. These texts are generated daily and, for this reason, it is common for people to want to know what are the current issues and their derivations. In this work, we propose to implement a mobile application that provides information to people, such as a degree of positive or negative polarity, on any relevant topic in society, helping in this way, so that people can make the best decision. The application will use several text classification techniques together. These techniques are focused on machine learning and lexical approaches.

Keywords: Sentiment Analysis; Twitter; Lexical, Machine Learning

Lista de Figuras

Lista de tablas

Lista de acrónimos

AS: Análisis de Sentimientos

API: Application Programming Interface

DT: Decision Tree

IMDb: Internet Movie Database

KNN: K Neighbours Nearest

ML: Machine Learning

NB: Naïve Bayes

NN: Neural Network

PLN: Procesamiento de Lenguaje Natural

RB: Red Bayesiana

RF: Random Forest

SEPLN: Spanish Society for Natural Language Processing

SemEval: Semantic Evaluation

SVM: Support Vector Machine

TASS: Taller de Análisis Semántico en la SEPLN

Tabla de Contenidos

1. Introducción

1.1. Motivación

El crecimiento de la comunicación, así como en la cantidad de información, ha permitido que muchos microblogs tengan un impacto en la sociedad. Uno de los principales microbloggings es Twitter, donde algunas personas de todo el mundo pueden escribir hasta un máximo de 280 caracteres denominado tweet, el cual puede ser compartido con más personas dentro de la red social. Teniendo en cuenta la gran cantidad de tweets que se mencionan en esta red social, podemos hacer uso de este recurso para el análisis de sentimientos, con el fin de saber, que es lo que algunas personas piensan sobre un determinado producto, temática, o una problemática social entre otros, esto con el fin de tener una visión general y así poder tomar una mejor decisión, frente algún problema que se suscite. La tarea de clasificación a gran escala es muy complicada, por lo que es necesario utilizar procesos automatizados que hagan este trabajo. Entre los procesos usados tenemos a *machine learning* o aprendizaje automático, que brinda un análisis a un conjunto grande de datos a partir de un conocimiento previo. Considerando esto, muchos trabajos [1]–[3] mencionan a *machine learning* como una herramienta en la clasificación de texto.

En las últimas décadas, la clasificación de texto ha llamado el interés de una gran cantidad investigadores, dentro del campo de la minería de datos. Esto debido a la necesidad de extraer información valiosa de forma eficaz y automatizada para los sectores empresariales y particulares. Pero la minería de datos no es el único mecanismo para lograr este objetivo, sino que se introducen conceptos específicos tales como la minería de opinión relacionada con el análisis de sentimientos [4].

El análisis de sentimientos en Twitter ha sido motivo de muchas investigaciones [5]–[11], ya que presenta un impacto social sobre las personas, esto debido a los tweets escritos en la red social. Teniendo presente cada tweet podemos decir que es una expresión propia de cada persona sobre algún tema de interés, estas expresiones escritas en los tweets pueden ser analizadas sentimentalmente. Además es importante destacar que siempre existe un nivel de dificultad al trabajar con el concepto de sentimientos [12], ya que no se puede relacionar

todas las maneras en las que las personas pueden expresar sus opiniones. Por ejemplo, una persona puede expresar una opinión como: '¡Éste es un gran producto!'; que es una opinión positiva, mientras que otro podría decir '¡Uno podría pensar que este es un gran producto!'. En esta oración, la frase "podría pensar" ha cambiado todo el significado de la opinión expresada; además, hay una gran dificultad para diferenciar entre hechos y opiniones en un texto e identificar la subjetividad oculta entre líneas.

El análisis de sentimientos hoy en día es utilizado por muchas empresas famosas como son: Facebook, Google, Twitter, etc. Muchas de ellas obtienen un valor agregado, además de presentar una información a sus clientes o usuarios. Dentro de las áreas de aplicaciones del análisis de sentimientos podemos destacar: Marketing, Política, Educación, etc.

En Marketing, dentro del análisis de sentimientos las empresas pueden realizar el monitoreo de sus campañas publicitarias, en base a este análisis las empresas pueden garantizar su toma de decisiones, de una manera técnica.

En Política, el análisis de sentimientos puede llegar hacer una herramienta predictora en un momento electoral de la sociedad. Permitiendo así tener resultados preliminares antes de que las elecciones tomen lugar.

En la educación, el análisis de sentimientos puede brindar ayuda con la detección de maltratos o discriminación en las instituciones educativas como: las escuelas, colegios, etc, y de esta manera evitar abusos o eventos a posterior.

El aumento en la cantidad de información, así como de los dispositivos móviles (Smartphones) ha concedido que las personas estén conectadas al Internet y tengan conocimiento de los temas actuales que acontecen en ese momento, esto a su vez ha permitido que se pueda realizar el análisis de sentimientos con la gran cantidad de información que se dispone. Por este motivo en el presente trabajo se desarrolla un aplicativo móvil enfocado hacia el análisis de sentimientos mediante la información provista por Twitter.

1.2. Objetivos

- Hacer el estudio de preprocesamiento de tweets, con sus diferentes procesos para el tratamiento del mismo.
- Realizar un estudio de los métodos de *machine learning* para el análisis de sentimientos en la red social Twitter.
- Hacer un estudio con las herramientas estadísticas usadas en la implementación del análisis de sentimientos en Twitter.
- Desarrollar un aplicativo móvil para el análisis de sentimientos en la red social Twitter, la cual presenta la polaridad frente a una temática definida o hacia una palabra consultada por el usuario final.

1.3. Esquema del documento

En esta sección se presentará la estructura de este documento.

En el Capítulo 2 se menciona los fundamentos necesarios para abordar el tema de análisis de sentimientos, así como el estado del arte, donde se presentan las técnicas usadas actualmente.

En el Capítulo 3 se menciona la propuesta que se va a realizar en el presente trabajo de disertación; contempla aspectos técnicos como son: las características, requerimientos funcionales, requerimientos no funcionales del aplicativo móvil a desarrollar.

En el Capítulo 4 se realiza la implementación correspondiente, a la propuesta definida en el capítulo anterior.

En el Capítulo 5 se presenta las pruebas realizadas con el aplicativo móvil desarrollado, así como el análisis de los resultados que se presentan.

Por último, en conclusiones y trabajo futuro, se presentan las conclusiones obtenidas en el trabajo de disertación, estructuradas por capítulos y de manera general. En el último capítulo se presentan las líneas de trabajo futuro que se abren con esta disertación y que están relacionadas con el análisis de sentimientos en Twitter.

2. Fundamentos y Estado del Arte

2.1. Introducción

En este capítulo se van a mencionar los fundamentos necesarios para el análisis de sentimientos acerca de la red social Twitter, dentro de los cuales se encuentran: el preprocesamiento de texto, extracción de características, clasificadores y la evaluación de clasificadores; además, en el capítulo se abordará las técnicas y procesos que actualmente se usan para el análisis de sentimientos en Twitter.

2.2. Redes Sociales

Las redes sociales se definen con el conjunto de herramientas digitales que respaldan la comunicación y la interacción entre los usuarios. Estas herramientas digitales como: wikis, blogs y microblogging entre otras, facilitan la creación y el intercambio de contenido y permiten la comunicación de muchos a muchos. El término red social se atribuye a los antropólogos británicos Alfred Radcliffe-Brown y Jhon Barnes[13]. Las redes sociales son parte de nuestras vidas, además de que las personas mantienen vínculos o conexiones que existen antes de las redes sociales y del Internet.

Actualmente las redes sociales son usadas por muchas personas alrededor del mundo, entre las redes sociales más populares podemos hacer mención a: Facebook, MySpace, Twitter, LinkedIn y sitios web similares, que alientan a sus usuarios a crear una lista de: amigos, seguidores y contactos que pueden generar conexiones indirectas a otros. Este tipo de sitios intenta de formas diferentes tomar en serio el viejo adagio de que “no es lo que usted conoce sino a quién conoce”, es aquí donde se introduce el concepto de red con sus múltiples conexiones y que puede proporcionar ayuda, apoyo, oportunidades e incluso dar una sensación de bienestar que de otra manera no sería posible.

En las posteriores secciones se trata a profundidad algunos temas relacionados a las redes sociales, tales como: la historia de las redes sociales, su seguridad, privacidad y finalmente se mencionará la red social Twitter.

2.2.1. Historia de las Redes Sociales

Mencionar la historia de las redes sociales es una tarea difícil, ya que el origen es difuso, además de que su evolución se ha venido dando en forma acelerada. Por esta razón se mencionará cronológicamente en la *Tabla 1* los eventos más transcendentes dentro de las redes sociales, como por ejemplo el primer email enviado entre dos ordenadores, así como el nacimiento de las redes sociales [14]. En la *Tabla 1* tenemos dos columnas: la primera son los años y la segunda, se presenta una pequeña descripción correspondiente a ese año.

Año	Descripción
1971	Envió del primer email entre dos ordenadores.
1978	Ward Christensen y Randy Suess crean un BBS (Bulletin Board Systems) para notificar, publicar y compartir información con sus amigos.
1994	Se lanza *GeoCites*, permitiendo a las personas desarrollar sus propios sitios web y alojarlos.
1997	Se lanza AOL Instant Messenger permite a los usuarios el chat, además que comienza el blogging y se lanza Google.
1998	Nace *Frieds Reunited*, es una red social de Reino Unido muy parecida con Classmates.
2000	Se llega a la cifra de 70 millones de ordenadores conectados.
2003	Nacen LinkedIn, MySpace y Facebook.
2006	Se lanza la red social de microblogging Twitter.
2010	Facebook cuenta con 550 millones de usuarios. Twitter registra diariamente 65 millones de tweets.
2012	Facebook supera los 800 millones de usuarios y Twitter cuanta con 200 millones.

Tabla 1. Cronología de las Redes Sociales

2.2.2. Seguridad y Privacidad

Tanto la confidencialidad como la privacidad tienen una gran importancia en las redes sociales desde la divulgación, así como el mal uso de información personal, la cual puede causar perjuicios con la vida de las personas involucradas. La privacidad en contexto de redes sociales tienes varias categorías [15].

- Anonimato de identidad del usuario: la protección de identidad de un usuario, así como los cambios de identidad en diferentes tipos de redes sociales pueden variar. No hay anonimato de identidad de usuario ya que muchas aplicaciones de Facebook dependen de la conexión con su perfil, así como de sus identidades públicas. En portales de citas como Friendster, el usuario crea un seudónimo débil al hacer solo el primer nombre de un participante visible para los demás, y no su último nombre [15].
- Privacidad del espacio personal del usuario: la visibilidad del perfil de usuario puede variar de una red social a otra, MySpace permite a los usuarios elegir si su perfil sea público o solo para amigos , Facebook en cambio permite que los usuarios de una subred puedan ver el perfil, a menos que el propietario haya restringido este acceso [15].
- Privacidad de comunicación del usuario: adicional a los datos personales que un usuario puede divulgar en el espacio digital, también el usuario puede divulgar información personal al operador de red o la red social. En estos datos puede haber la hora de conexión, longitud, latitud, dirección IP entre otras.
- Autenticación e integridad de los datos: en [16] se menciona que la mayoría de redes sociales respaldan sus relaciones sociales preexistentes con la vida real. Teniendo estos datos almacenados en la red social se puede modelar con un gráfico social.

Dentro de las redes sociales más populares y microbloggings con gran incidencia o impacto en la sociedad, tenemos a Twitter. La cual será detallada en la siguiente sección, ya que se considerará como una pieza fundamental para el desarrollo del trabajo.

2.2.3. Twitter

Los orígenes de Twitter se dan en el año 2006 donde es creado por Biz Stone, Evan Williams y Jack Dorsey [17]. En un principio se denominó Twtrr. Es una aplicación de microblogging gratuita, presenta algunas ventajas de blogs, redes sociales y mensajería instantánea que permite compartir: opiniones, pensamientos, enlaces con otras personas o usuarios de una forma pública o privada. La empresa tiene su sede en San Francisco, California, con filiales en San Antonio (Texas) y Boston (Massachusetts) en Estados Unidos; a continuación, se presenta la *Figura 1* con el logotipo de la empresa.

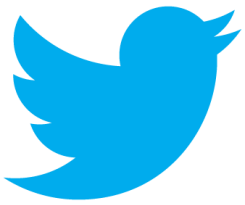

Figura 1. Logotipo de Twitter

Una de las principales características de Twitter es el límite de caracteres que se puede escribir en cada tweet, este límite de 280 caracteres fue lanzado recientemente el 26 de septiembre del 2017, anteriormente se tenía un límite de 140, estos mensajes son visibles en la página principal del usuario. Cuando uno de los usuarios de Twitter ingresa a su cuenta, en donde se le presenta un listado de manera cronológica de los tweets de otros usuarios, así como se muestra la *Figura 2*, de igual forma el usuario puede transmitir los tweets a los usuarios que le siguen:

Figura 2. Cronología de los tweets y límite de caracteres[1]

La información mostrada en Twitter por lo general es pública, aunque en algunos casos los usuarios pueden restringir el acceso a su información. Por esta razón Twitter, es una de las redes sociales más utilizadas para obtener información. Para la extracción de tweets, la red social Twitter ha desarrollado un API en el cual permite obtener esta información, y así poder ser analizada en un determinado punto de vista.

Los tweets cuentan con ciertos términos en particular propios de esta rede social, entre ellos tenemos:

- **RT o retweet:** Es una acción en la cual un usuario, seguidor o no, puede reenviar un tweet a otro usuario.
- **Usuario:** Es un miembro y gestor de su cuenta en la red social. Cada usuario puede referirse a otro utilizando un @ antes del nombre de usuario.
- **Time Line (TL):** Es una pista del tweet o cronología de un mensaje enviado en la red social. Los mensajes se presentan en un orden según son escritos, de esta forma el más reciente se encuentra más arriba.

[1] https://blog.twitter.com/official/en_us/topics/product/2017/Giving-you-more-characters-to-express-yourself.html

- **Seguidores o followers:** Son personas que mantienen una cuenta en la red social y a su vez siguen a otra cuenta.
- **Seguidos o following:** Son las cuentas de usuario que siguen a cuentas de otros usuarios de la red social.
- **Favorito:** Es una acción propia del aplicativo para elegir tweets como favoritos.
- **Mención:** Es una acción en la que se nombra a un usuario en el TL.
- **DM o direct message:** Esta funcionalidad permite enviar mensajes privados a aquellos usuarios que sigan la cuenta.
- **Hashtag o etiqueta:** Se usa para marcar temas, conceptos, términos y se lo representa con un #. Una vez que se marca antes de una palabra, se puede buscar todos los tweets que contengan este símbolo # más la palabra.

Frente al crecimiento global de la red social Twitter; a continuación se muestra la *Figura 3*, donde se destacan las cuentas activas de Twitter para el año 2014, por regiones a nivel mundial. Es aquí donde podemos ver que América Latina presenta un valor del 1% mayor a las cuentas activas de Norte América y solo siendo superado por África con un 29%.

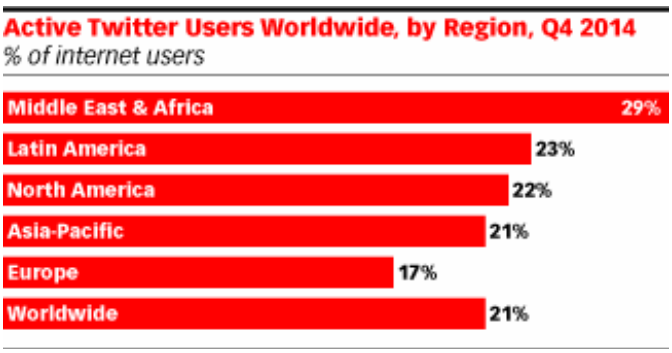

Figura 3. Usuarios activos de Twitter en el mundo[2]

Como información destacable, ya para inicio del 2012, en donde Chile se ubica en el puesto 19 en Latinoamérica con una población muy parecida a la de Ecuador con 18 millones de personas y un valor cercano a 5 millones de cuentas como se muestra en la Figura 4. De los 5 millones de cuentas en este país, el 24% son cuentas activas (1,2 millones), es decir, que

[2]https://www.emarketer.com/Article/Latin-America-Register-Highest-Twitter-User-Growth-Worldwide-2015/1012498

han escrito un tweet o más en los últimos 3 meses, superando a países como Alemania, Filipinas e India, lo que se muestra en la *Figura 5*.

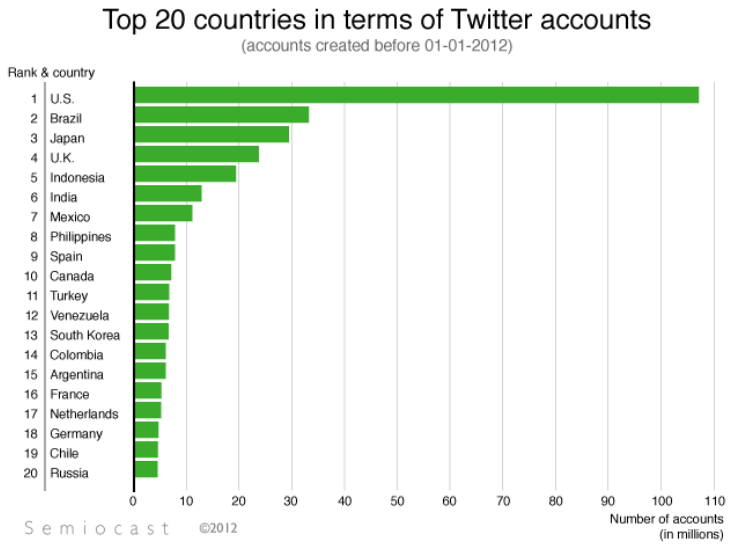

Figura 4. Número de cuentas por países en millones[3]

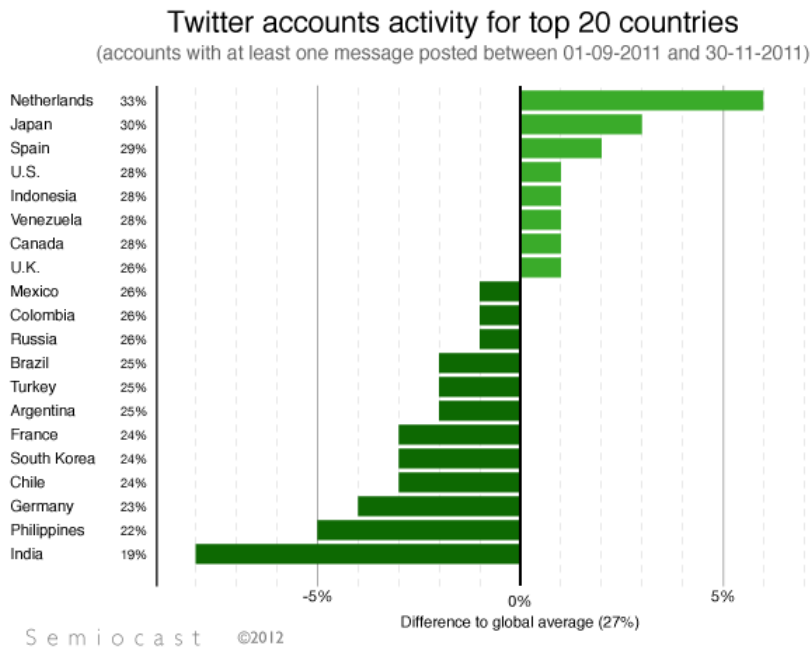

Figura 5. Porcentaje de cuentas activas por país[4]

[3]https://semiocast.com/en/publications/2012_01_31_Brazil_becomes_2nd_country_on_Twitter_superseds_Japan

[4]https://semiocast.com/en/publications/2012_01_31_Brazil_becomes_2nd_country_on_Twitter_superseds_Japan

Después de presentar la información del crecimiento de la red social Twitter, podemos destacar que los países de Latinoamérica y en particular Ecuador, tienen un impacto en el crecimiento de esta red social, esto debido a los usuarios, que indirectamente la usan para expresar sus comentarios, ideas, críticas, etc. Es por esta razón que muchos estudios utilizan Twitter para analizar o determinar el comportamiento de la sociedad, frente alguna problemática. Dentro de los estudios relacionados con Twitter tenemos al análisis de sentimientos [18]–[20], el cual se mencionará con mayor detalle en la siguiente sección.

2.3. Análisis de Sentimientos

Debido al desarrollo global de Internet a inicios de los años 90, se ha disminuido las fronteras entre los consumidores y los medios de comunicación. Hoy en día nos encontramos en el escenario de la Web 2.0, en la cual se enfoca que no solo los usuarios consumen la información, sino también ellos son participes para la generación mediante sus comentarios, críticas, debates y opinando en las redes sociales.

De este suceso nace el Análisis de Sentimientos (AS) bajo una idea de procesar o analizar las opiniones de una forma automática para obtener un valor fundamental y toma de decisiones adecuada, mediante el Procesamiento de Lenguaje Natural (PLN) [21]. El AS o también denominado Minería de Opinión, es área muy extensa que no solo se enfoca en la detección de polaridad de texto, comentarios u opiniones, si no que va más allá como es la detección de emociones, extracción de características en un texto, a un nivel de entidad o de evaluar la reputación de una empresa entre otras muchas funciones.

Con respecto a la detección de polaridad el AS intenta determinar la orientación sentimental de un texto, el cual puede ser positivo, negativo o neutro referente a un tema. Además, este análisis va más allá del concepto de polaridad, tratando de identificar el estado emocional de una palabra, como la ira, la tristeza, la felicidad, etc. [22].

La hipótesis principal del AS a nivel de documento/comentario/tweet menciona que dado un conjunto de tweets/documentos D, se determina que cada documento perteneciente al conjunto $[d \in D]$ expresa una opinión tanto positiva o negativa según un objeto único y que aquella opinión pertenece a un único sujeto [23].

Actualmente hay tres diferentes niveles para realizar la clasificación:

- **Nivel documental:** consiste en obtener una opinión global del documento, como es positiva, negativa o neutral.
- **Nivel de sentencia:** tiene como objetivo hallar el sentimiento a nivel de frase individualmente. En este caso es primordial conocer si la frase es subjetiva u objetiva.
- **Nivel de aspecto:** tiene como objetivo buscar el sentimiento acorde a cada sujeto dentro de la propia frase. En este caso es primordial reconocer los aspectos o sujetos dentro de cada sentencia.

Para realizar el AS es necesario considerar ciertas fases o pasos con el propósito de obtener un resultado final que podría ser la polaridad de un comentario, para esto a continuación se muestra la *Figura 6* con los pasos comúnmente usados en varias investigaciones [24]–[27], para el análisis de sentimientos en Twitter.

Figura 6. Estructura general del análisis de sentimientos

En las siguientes secciones se detallarán cada una de las fases usadas en el análisis de sentimientos, tales como: adquisición de datos, preprocesamiento de texto, selección de características y clasificación, sobre la red social Twitter.

2.3.1. Adquisición de Datos

El proceso de análisis de sentimientos necesita de datos para comenzar el análisis, por este motivo es importante adquirir ya sea de una fuente externa como los que se menciona en la *Tabla 2* , donde podemos observar la fuente de datos, de que tipo son los datos, la referencia tomada y una pequeña descripción del *dataset*.

El uso de un *dataset* público es una alternativa para empezar el análisis de sentimientos, pero también podemos obtener datos, mediante el uso de un API, en el caso de Twitter podemos utilizar su API para la extracción de datos.

Fuente	Tipo	URL	Descripción
Universidad Stanford, estudiantes graduados de Ciencias de la Computación.	Marcas, productos y de un tema sobre Twitter.	http://help.sentiment140.com/for-students/	Entrenamiento contienen 1600000 y de prueba contienen 497 sentencias.
Universidad de Michigan	Comentarios	https://www.kaggle.com/c/si650winter11	Entrenamiento contienen 7086 y de prueba contienen 33052 sentencias.
Spam dataset	Revisiones de correo no deseado	http://myleott.com/op_spam	400 críticas engañosas y 400 verídicas en la categoría positiva y negativa.
Soe dataset	Sarcasmo y comentarios desagradables	http://nlds.soe.ucsc.edu/iac	1,000 discusiones, 390,000 publicaciones, y algunas 73,000,000 palabras
Stanford dataset	Movie reviews	http://ai.stanford.edu/~amaas/data/sentiment/	50000 críticas de películas
TASS 2012	Tweets	http://www.sepln.org/workshops/tass/2012/corpus.php	70000 tweets, escritos en español, relacionado con personalidades, economía y política.
IMDb	Movie reviews	https://www.cs.cornell.edu/people/pabo/movie-review-data/	2000 comentarios entre positivos y negativos
Twitter2016-train	Tweets	http://alt.qcri.org/semeval2016/	6000 tweets entre positivos, negativos y neutros
twitter2016-dev	Tweets	http://alt.qcri.org/semeval2016/	2000 tweets entre positivos, negativos y neutros
twitter2016-test	Tweets	http://alt.qcri.org/semeval2016/	20632 tweets entre positivos, negativos y neutros
twiiter2017-test	Tweets	http://alt.qcri.org/semeval2017/	12284 tweets entre positivos, negativos y neutros
SMS2013	Tweets	http://alt.qcri.org/semeval2014/task9/	2093 tweets entre positivos, negativos y neutros
Twt2014-sarcasm	Tweets	http://alt.qcri.org/semeval2014/task9/	86 tweets entre positivos, negativos y neutros
LiveJournal2014	Tweets	http://alt.qcri.org/semeval2014/task9/	1142 tweets entre positivos, negativos y neutros
Semval 2015	Tweets	http://alt.qcri.org/semeval2015/	65854 tweets
SS-Twitter	Tweets	https://onlinelibrary.wiley.com/doi/abs/10.1002/asi.21662	4242 tweets

Tabla 2. Listado de datasets

Los *dataset*s que fueron mencionados en la Tabla *2* destacan en varios trabajos de investigación vinculados al análisis de sentimientos, algunos de ellos se presentan en idioma ingles y otros en español. Los que destacan en idioma español se enfocan sobre España, referente a temas políticos, culturales, económicos, entre otros. Considerando esto, no se podría implementar dentro del presente trabajo estos *datasets*, ya que el idioma inglés sería un limitante, así como los *datasets* en español, que son únicamente enfocados a España.

La adquisición de datos es un paso muy importante para el análisis de sentimientos, ya que sin datos no daría inicio el análisis. Durante la adquisición se debe considerar el uso de un *dataset*, teniendo en cuenta ciertas características como, por ejemplo: el idioma, sentimientos, temática, etc. Estas características del *dataset*, dependerá del análisis que se vaya ha realizar. A continuación, se detallará el preprocesamiento de texto al *dataset*.

2.3.2. Preprocesamiento

Dentro de las fases del análisis de sentimientos tenemos el preprocesamiento de texto [28]. En esta fase tenemos varias técnicas que se detallará a continuación, estas técnicas reducen el ruido de los comentarios, también permiten la reducción de dimensiones y una selección correcta de los datos que posteriormente serán usados en la siguiente fase como es la extracción de características.

- *Limpieza de Datos*

La limpieza es una tarea para eliminar el ruido posible de los datos generados en Twitter que fueron, por medio de la eliminación de cierto contenido que no aporta significativamente con sentimientos; a continuación, se detalla las técnicas para la eliminación del contenido innecesario para el análisis de sentimientos:

  - *Añadir etiquetas*

En Twitter, muchos usuarios hacen uso de símbolos como son: signos de pregunta, signos de exclamación, emoticones, etc. El análisis individual de estos caracteres se hace muy difícil por esta razón se trasforma en etiquetas representativas para el análisis, este proceso es usado en [20]; a continuación, se presenta la *Tabla 3* con un ejemplo de símbolos y sus correspondientes etiquetas:

Símbolos	Etiquetas
”) :“ ,”)-:“ ,”):“	Feliz
”(:“ ,”(-:“ ,”(:“	Triste
!	Exclamación
?	Pregunta

Tabla 3. Ejemplos de símbolos con sus respectivas etiquetas

- *Eliminar Números*

Es muy frecuente la eliminación de caracteres numéricos en los comentarios de Twitter, aunque muchos investigadores han mencionado que los números puede mejorar la eficiencia de la clasificación de texto[29].

- *Eliminar Stopwords*

Los *stopwords* son palabras usadas con mucha frecuencia en las oraciones, así como las que se muestran en la *Tabla 4*. En algunas investigaciones [12], [25] se realiza la eliminación de estas palabras, ya que no aportan en gran medida al análisis de sentimientos y puede causar ruido al conservarlas; a continuación, se muestra la *Tabla 5* presentando una lista de repositorios públicos de *stopwords*.

Palabras			
un	ir	podría	cierto
una	tengo	yo	desde
sobre	el	usan	conseguir
todo	la	estoy	están
algún	donde	vamos	porque

Tabla 4. Ejemplos de stopwords

No	Descripción	URL
1	Lista de Stopwords en Ingles.	http://xpo6.com/list-of-english-stop-words/
2	Lista de Stopwords en Ingles.	https://www.ranks.nl/stopwords
3	Lista de Stopwords en español.	https://www.ranks.nl/stopwords/spanish
4	Lista de Stopwords en Ingles.	https://algs4.cs.princeton.edu/35applications/stopwords.txt

Tabla 5. Lista de sitios públicos con stopwords

- *Eliminar signos de puntuación*

En muchos trabajos, es muy frecuente la eliminación de signos de puntuación en el procesamiento de texto [29]. Sin embargo, se debe considerar que la presencia de signos de puntuación denota algún sentimiento. Como, por ejemplo, una exclamación podría significar un sentimiento positivo o negativo. Considerando esto nosotros podríamos eliminar los signos de puntuación afectando con la exactitud de la clasificación.

- *Lowercasing*

Esta técnica de preprocesamiento de texto es muy usada ya que permita transformar todas las palabras de un tweet desde mayúsculas a minúsculas, al hacerlo muchas palabras se fusionarán y la dimensionalidad se reducirá, ya que una palabra será considerada como una sola entrada, independientemente si es mayúscula.

- *Lemmatizing*

Lemarizacion es otra técnica de preprocesamiento de texto la cual permite unir las palabras, mediante la eliminación de terminaciones en las palabras, esto con el fin de encontrar sus lemas o su forma raíz en un diccionario. Permitiendo de esta manera que las palabras se fusionen y la dimensionalidad se reduzca.

- *Stemmer*

Esta técnica permite eliminar las terminaciones de las palabras y detectar la forma raíz de las mismas, al hacerlo muchas palabras se fusionan y la dimensionalidad se reduce. Es un método ampliamente utilizado que generalmente proporciona buenos resultados en la

clasificación de documentos; a continuación, se presenta la *Tabla 6* con listado de palabras en su forma normal y su correspondiente forma raíz.

Palabra normal	Stemming
departamento	Departamento
desarrollo	Desarroll
plan	Plan
educación	Educ
años	Años
base	Bas
unas	Unas
personas	Person
programa	Program
partido	Part
parte	Part
partida	Part
partes	Part

Tabla 6. Lista de palabras con su forma raíz

- *Reemplazo de Abreviaciones*

Muchos usuarios en Twitter hacen uso de abreviaciones al momento de escribir sus tweets, esto al momento de realizar el análisis de sentimientos presenta un problema, por esta razón esta técnica permite reemplazar estas abreviaciones por su significado respectivamente.

- *Reemplazo de Contracciones*

Los usuarios de Twitter en muchas ocasiones al escribir tweets, recortan las frases o hacen uso de contracción como es el caso en el lenguaje Ingles, esta técnica permite reemplazar las contracciones por sus palabras equivalentes, como, por ejemplo: "*won't*" en este caso se reemplazaría por "*will not*".

- *Remplazo de URL y Nombres de Usuarios*

En Twitter casi siempre el contenido de los tweets cuenta con URLs y nombres de usuarios, esta información no es relevante en temas de análisis de sentimientos, por esta razón esta

técnica permite reemplazar tanto las URL como los nombres de usuarios por palabras como "URL" y "USERNAME".

Dentro de las investigaciones encontradas, tenemos en [30], donde se menciona una mínima diferencia de resultados si son suprimidos los *stopwords*, ya que los tweets son textos cortos y tal vez con textos más amplios se notaría su repercusión. Sin embargo en [31] se realiza una eliminación de *stopwords*, que permite reducir las dimensiones de los términos y es beneficioso.

En la técnica *stemming* no existe un criterio generalizado, así como se menciona en la investigación [32], donde se descarta la técnica *stemmnig* durante el preprocesamiento de texto. Sin embargo en la investigación [30] es un factor que ayuda positivamente en la precisión del clasificador, correspondiente a los casos posibles. Por otro lado en [33] se aplica el preprocesamiento de datos usando *stemming*, especialmente el algoritmo de Porter, permitiendo así una reducción de dimensional de los datos, pero sutilmente. Ante la variedad de opiniones es probable que ambas opciones sean válidas, para ciertos corpus sea ventaja y para otra desventaja.

En el estudio "*Sentiment Analysis and Topic Detection of Spanish tweets*" [34] se usa las técnicas de procesamiento de datos como stemming y lemmatizacion mediante una librería de C++ "*FreeLing Software*". Como se menciona en este estudio el idioma español e inglés tienen diferencias considerables. El español es un idioma más flexible con relación al inglés por que las técnicas tales como *stemming* y lematizacion tendrán una mayor influencia en español que en el inglés. En el presente estudio se ha usado el corpus de TASS 2012 mencionada en la *Tabla 2*.

En [35] son implementadas algunas de las técnicas de preprocesamiento de texto sobre los tweets. Entre las técnicas usadas por ellos tenemos: eliminación de dígitos, eliminación de caracteres alfanuméricos, eliminar URLs, eliminar espacios en blanco, reemplazar emoticones por palabras, remover *stopwords*, reemplazo de acrónimos y manejo de negaciones. Estas técnicas fueron aplicadas hacia tweets previamente extraídos con el API de Twitter. Por otro lado, en [27] se realizó el preprocesamiento de texto sobre el *dataset* "*Internet Movie Database* (IMDb)" mencionado en la *Tabla 2*. Las técnicas empleadas fueron la división de palabras como *tokens*, para posteriormente usar *stemming* y reducir el número de *tokens*, adicionalmente a esta técnica, ellos implementaron la eliminación de *stopwords*. Los resultados que obtuvieron al usar estas técnicas fueron favorables al

momento de realizar el análisis de sentimientos, por este motivo destacan así la importancia que tiene la fase de preprocesamiento texto, así como las técnicas que son usadas.

En [36] se detallan varias técnicas de preprocesamiento de texto, entre las que son usadas tenemos: la tokenizacion de los tags HTML, *username*, URLs, *lowercase* y eliminación de *stopwords*. Las pruebas realizadas en esta investigación se dieron en varios *dataset* como: Twitter2016-train, twitter2016-dev, twiiter2017-test, SMS2013, Twt2014-sarcasm y LiveJournal2014, estos *datasets* son mencionados en la *Tabla 2*.

En [37] se menciona el uso de las siguientes técnicas de preprocesamiento de texto: remover URLs , eliminación de *stopwords*, eliminación de números, expandir los acrónimos sobre los métodos de *machine learning* como: *Logical Regression*, *Naïve Bayes*, *Support Vector Machine, Random Forest*. En esta investigación los resultados experimentales indican que la eliminación de URL, eliminación de números afectan mínimamente al rendimiento de los clasificadores; además, reemplazar la negación y expandir los acrónimos puede mejorar la precisión de la clasificación. Por lo tanto, eliminar *stopwords*, números y URL es apropiado para reducir el ruido, pero no afecta el rendimiento.

La fase de preprocesamiento de texto en tweets es muy importante, ya que tiene como objetivo realizar la limpieza de los datos, permitiendo así una reducción de dimensionalidad haciendo que el problema se simplifique, además de eliminar el ruido de ciertos datos que no contribuyen directamente en el análisis de sentimientos. Para esto se puede implementar las técnicas de preprocesamiento de texto mencionadas en la sección 2.3.2.

Una vez realizado el preprocesamiento de texto en tweets, el siguiente paso es la selección de características, la cual se presentará en la siguiente sección.

2.3.3. Selección de Características

En esta etapa del análisis de sentimientos se abordará técnicas para la representación de los documentos o tweets, ya que los métodos supervisados de *machine learning* requieren de una representación, como es de un vector de características ponderadas.

- *Representación del Tweet*

Los tweets que se generan diariamente en Twitter son no estructurados, por este motivo se debe tener una representación adecuada para su computación.

Dentro de los tweets o documentos hay palabras o características, cada tweet se muestra como un vector en el espacio o también como (BoW "Bag of Words"); cada dimensión del espacio representa una característica del tweet. Los tweets son puntos en un espacio dimensional. Los tweets se representan como un vector en el espacio $d_i = (w_{i1}, w_{i2}, \ldots\ldots, w_{ik}, \ldots\ldots, w_{i|w|})$. Finalmente se mapea los documentos o tweets hacia una matriz llamada "términos" por la matriz de documentos o tweets, esta representación corresponde al espacio de características[38]; esta representación se muestra en la *Figura 7*.

	w_1	w_2	…	w_k	…	$w_{\|w\|}$
d_1	w_{11}	w_{12}	…	w_{1k}	…	$w_{1\|w\|}$
d_2	w_{21}	w_{22}	…	w_{2k}	…	$w_{2\|w\|}$
…..	…	…	…	…	…	…
d_i	w_{i1}	w_{i2}	…	w_{ik}	…	$w_{i\|w\|}$
…..	…	…	…	…	…	…
$d_{\|D\|}$	$w_{\|D\|1}$	$w_{\|D\|2}$	…	$w_{\|D\|k}$	…	$w_{\|D\|\|w\|}$

Figura 7. Representación del espacio de características [38]

La representación binaria de tweet se puede presentar por la ausencia o presencia de una característica $w_{|D||w|}$, esto puede estar representado en forma numérica como: 1 presencia y 0 ausencia, esta es una forma de representación, pero existen diversas técnicas para la extracción de características; a continuación, son detalladas:

- *TF-IDF*

Dentro de la colección de documentos, el valor de Frecuencia de Términos (TF = *Term Frequency*) o Frecuencia Inversa de Términos (IDF = *Inverse Term Frequency)* aumenta con el aumento en la frecuencia de una palabra en particular en el documento; a continuación, se presenta la *Figura 8* con un ejemplo, el cual representa la frecuencia de términos:

Figura 8. Frecuencia de términos en el tweet

El concepto de frecuencia de términos es el número de veces que aparece un término particular en el texto y la IDF mide la frecuencia de ocurrencia de cualquier palabra en todos los documentos y está dada por la siguiente formula [38]:

$$IDF(w_k) = \frac{|D|}{DF(w_k)}$$

Donde $|D|$ es número de documentos en la colección. DF es el número de documentos en colección en la que la palabra w_k aparece.

- *Información mutua*

La información mutua es una técnica de selección de características que no está uniformemente distribuida en clases de sentimiento porque son informativas sus clases y podemos ver que la información mutua da más importancia a solo unos pocos términos del documento [39].

$$IM(f,c) = \sum c \in C \sum_f P(f,c) \log \frac{P(f,c)}{P(f)P(c)}$$

Donde $P(f, c)$ representa la función de distribución de probabilidad conjunta, $P(f)$ y $P(c)$ representan la distribución de probabilidad marginal de f y c. c es clases positiva y negativa.

- *Chi-Square*

Esta técnica de selección de características *Chi-Square* mide el conteo observado y el recuento esperado y analiza la cantidad de desviación que se produce entre ellos [39].

$$\lambda^2(f,c) = \frac{N(WZ - YX)^2}{(W+Y)(X+Z)(W+X)(Y+Z)}$$

W, X, Y, Z representa las frecuencias, así como la presencia o ausencia de características en la muestra. W es el recuento de las muestras en las que la característica f y c ocurrieron juntas. $\mathrm{N} = \mathrm{W} + \mathrm{X} + \mathrm{Y} + \mathrm{Z}$, f representa la característica y c representa la clase.

- *Latent Semantic Indexing (LSI)*

Los métodos de selección de características permiten la reducción de datos correspondientes al conjunto original. Los métodos de transformación de características crean un conjunto más pequeño de características en función del conjunto original de características. LSI es uno de los métodos de transformación de características más famosos [40]. El método LSI transforma el espacio de texto en un nuevo sistema de eje, que es una combinación lineal de las características de la palabra original. Las principales técnicas son el análisis de componentes (PCA) se utilizan para lograr este objetivo [41]. Determina el sistema de ejes conservando el mayor número de información mediante la agrupación de características o términos. La principal desventaja de esta técnica es que es no supervisada, por tanto, la distribución de la clase es ciega.

- *Information Gain (IG)*

Esta técnica de *Information Gain* es una de las más usadas en el análisis de sentimientos, gracias a que determina las características relevantes para predecir la revisión, mediante el estudio de la presencia o ausencia de características en el documento.

$$IG(f,c) = -\sum_{c,c} P(c)\log P(c) + \sum_{f,f} P(f) \sum_{c,c} P(c|f)\log P(c|f)$$

$\mathrm{P(c|f)}$ es la probabilidad conjunta, la clase es c y la característica es f y $\mathrm{P(c)}$ denota la probabilidad marginal.

- *Gain Ratio*

Esta técnica se utiliza como un proceso iterativo donde seleccionamos conjuntos más pequeños de características en forma incremental. Estas iteraciones finalizan cuando solo queda un número predefinido de características [42]. *Gain Ratio* se utilizó por primera vez en el árbol de decisiones (C4.5) y emplea la normalización al puntaje de *information gain* al utilizar un valor de información dividido [43]. El valor de información dividida corresponde a la información potencial obtenida al dividir el conjunto de datos de entrenamiento D en particiones v, lo que resulta en v resultados en el atributo A:

$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} x \log_2 \frac{|D_j|}{|D|}$$

Donde el valor alto de *SplitInfo* significa que las particiones tienen el mismo tamaño (uniforme) y un bajo valor de *SplitInfo* significa que pocas particiones contienen la mayoría de las tuplas. Finalmente, *gain ratio* se define como:

$$Gain\ Ratio(A) = \frac{Information\ Gain\ (A)}{SplitInfo(A)}$$

- *Algoritmo Reif-f*

Esta técnica consiste en seleccionar características al azar, calculando sus vecinos más cercanos y optimizando un vector de ponderación de características para otorgar más importancia a las características que discriminan la instancia de vecinos de diferentes clases [44]. Finalmente, reif-f intenta evaluar sobre la estimación de pesos w_f, esto mediante las probabilidades por el peso y el ranking de características f.

$$w_f = P(Diferente\ valor\ de\ f | Instancias\ mas\ cercanas\ de\ un\ diferente\ clase) - P(Diferente\ valor\ de\ f | Instancias\ mas\ cercanas\ a\ la\ misma\ clase)$$

Dentro de las técnicas de selección de características, hay varias investigaciones relacionadas al análisis de sentimientos, así como es el caso [42] donde ellos realizan una comparativa con varias técnicas de selección de características tales como: *document frecuency, information gain, gain ratio, chi y reif-f algoritm,* esto con la finalidad de medir el desempeño para la clasificación en términos de *recall*, *precision* y *accuracy*. En esta investigación, *gain ratio* presentó los mejores resultados para una gran cantidad de selección de características sentimentales (más de 5000 características), mientras que las otras técnicas de selección de características dieron un rendimiento pobre en F1. Además, ellos destacan en su trabajo que el rendimiento del clasificador SVM depende del número de características representadas con la selección de texto.

Las técnicas apropiadas en la selección de características dentro del análisis de sentimientos han tenido un papel importante, al momento de identificar los atributos y el aumento de rendimiento por parte de los clasificadores, como lo menciona [45].En varios estudios se ha intentado solventar la problemática de selección de características, mediante el uso de métodos de selección [32]. Algunas de estas investigaciones se centran en características

simples, como: palabras simples, n-gramas como: bígamas y trigramas [46]–[48], o también se presenta la combinación entre ellas.

En otro estudio, se procedió con la comparativa de las técnicas de selección de características, como: *document frequency, information gain , gain ratio, chi y relief-f*, sobre los clasificadores: *naïve bayes, support vector machine*, *k nearest* [49]. Los resultados obtenidos en el estudio, muestran que la técnica *gain ratio* tiene el puntaje de *accuracy* más alto con 88% a 90% frente a las demás técnicas de selección de características.

La selección de características es un paso muy importante para realizar el aprendizaje automático o *machine learning*, ya que las técnicas empleadas en *machine learning* requieren un vector de características, el cual se generaría al implementar las técnicas anteriormente detalladas.

2.3.4. Clasificación

En esta sección se abordará las técnicas usadas para la clasificación de texto, entre ellas tenemos métodos por *Machine Learning* (ML) y Léxico. A continuación, se muestra la *Figura 9* con el detalle general de las técnicas para la clasificación de texto.

Los métodos de clasificación de texto que utilizan el enfoque ML, se pueden dividir aproximadamente en: métodos de aprendizaje supervisados y no supervisados. Los métodos supervisados hacen uso de una gran cantidad de documentos previamente etiquetados; los métodos no supervisados se usan cuando es difícil de encontrar estos documentos ya etiquetados [25].

El enfoque basado en léxico depende de la opinión que se utiliza para analizar el texto. Existen dos métodos en este enfoque: el enfoque basado en el diccionario que depende de buscar palabras clave de opinión y el enfoque basado en corpus, el cual comienza con una lista inicial de palabras de opinión para luego encuentra otras palabras en un corpus grande y ayudar a encontrar palabras con orientaciones específicas del contexto. Esto podría hacerse mediante el uso de métodos estadísticos o semánticos [25].

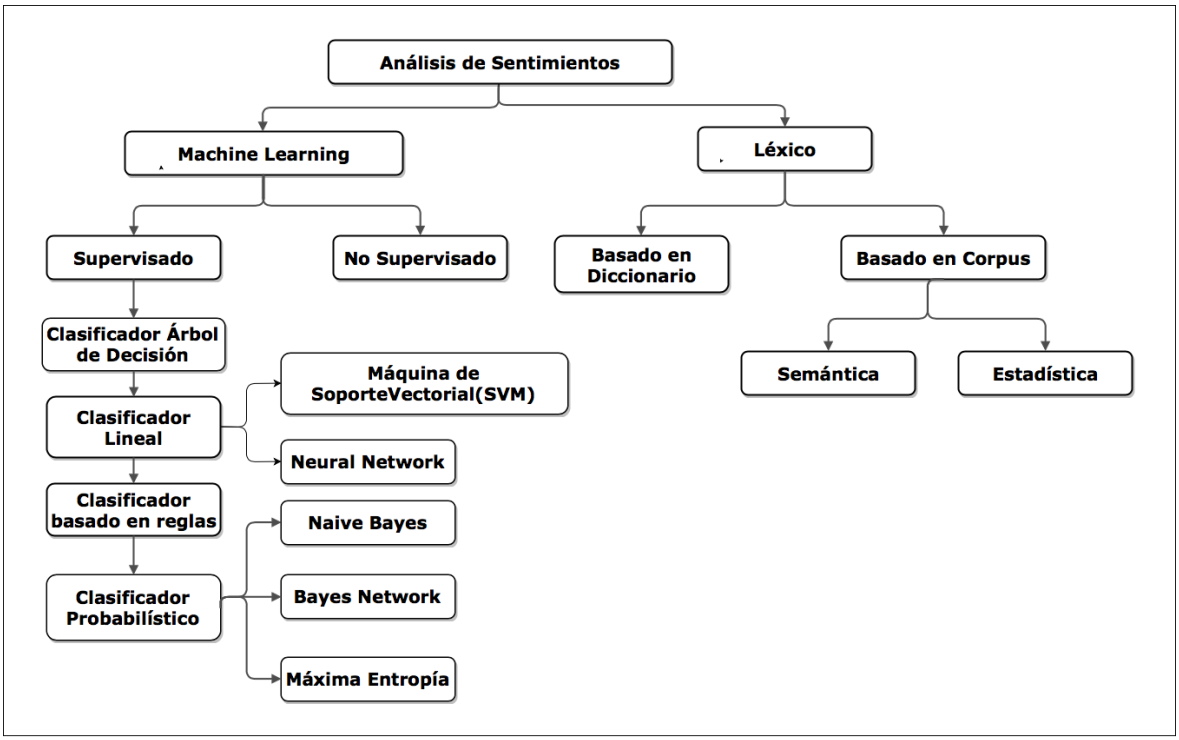

Figura 9. Técnicas de clasificación de sentimientos [24]

Machine Learning

El enfoque de aprendizaje automático o *machine learning* se basa en el uso de algoritmos que permiten el aprendizaje a las computadoras, a través de un conocimiento previo; en nuestro trabajo la problemática será para la clasificación de tweets mediante el uso de estos algoritmos.

Para la definición del problema de análisis de sentimientos en tweets, se ha definido un conjunto de entrenamiento $D = \{x_1, x_2, \ldots\ldots, x_n\}$ donde cada registro es etiquetado a una clase; el modelo de clasificación está relacionado con las características en el registro subyacente a una de las etiquetas de clase; luego, para una instancia dada de clase desconocida, el modelo se usa para predecir una etiqueta de clase.

Dentro de los métodos *machine learning* encontramos los supervisados y no supervisados:

- *Supervisado*

Los métodos de aprendizaje supervisado dependen de documentos de entrenamiento que contengan etiquetas, por lo general estas etiquetas las realiza un supervisor de manera manual; esto quiere decir que los documentos o tweets son catalogados o etiquetados según el supervisor. Posteriormente estos datos de entrenamientos son usados para la predicción

de la clase [26]; a continuación, se detallan los métodos más usados para el análisis de sentimientos:

- Clasificador Arboles de Decisión

El clasificador de árbol de decisión proporciona una división jerárquica del espacio de datos de entrenamiento en el que para dividir los datos [24][50], se usa una condición en el valor del atributo. La condición o predicado es la presencia o ausencia de una o más palabras. La división del espacio de datos se realiza recursivamente hasta que los nodos hoja contengan un número mínimo de registros que se utilizan para fines de clasificación.

- Clasificadores Lineales

  - Redes Neuronales

Las redes neuronales consisten en un gran número de unidades de procesamiento de información (llamadas neuronas) organizadas en capas, que trabajan al unísono. Puede aprender a realizar tareas (por ejemplo, clasificación) ajustando los pesos de conexión entre las neuronas, asemejándose al proceso de aprendizaje de un cerebro biológico [51].

  - Máquina de Soporte Vectorial (SVM)

El enfoque principal de las SVM es determinar los separadores lineales en el espacio de búsqueda que mejor pueden separar las diferentes clases. En la *Figura 10* se observa mejor la separación de clases mediante hiperplanos, los cuales proporcionan la mejor separación entre las clases; porque, la distancia normal de cualquiera de los puntos de datos es la más grande, por lo que representa el margen de separación máximo [52].

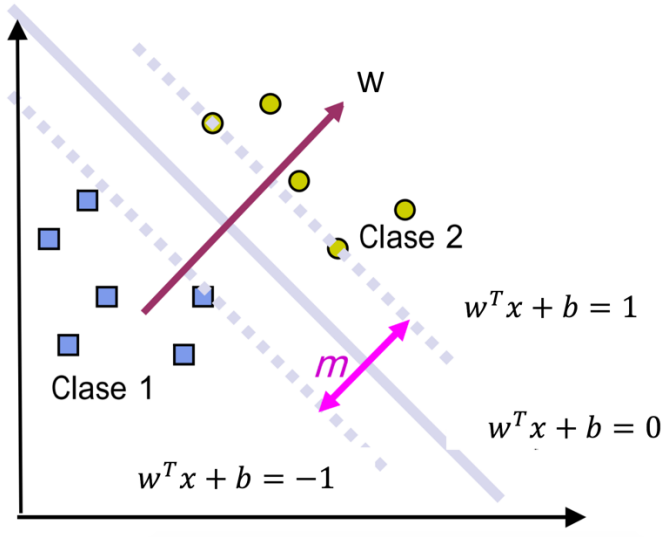

Figura 10. Clasificación de tipo SVM frente a un problema [52].

- Clasificador basado en reglas

En los clasificadores basados en reglas, el espacio de datos se modela con un conjunto de reglas. El lado izquierdo del espacio está representado por una condición en el conjunto de características expresado en forma disyuntiva normal, mientras que el otro lado comprende la etiqueta de clase.

- Clasificadores Probabilísticos

Los clasificadores probabilísticos usan modelos de mezcla para la clasificación. El modelo de mezcla supone que cada clase es un componente de la mezcla. Cada componente de mezcla es un modelo generativo que proporciona la probabilidad de muestrear un término particular para ese componente. Este tipo de clasificadores también se denominan clasificadores generativos. Tres de los clasificadores probabilísticos más famosos se discuten en las siguientes subsecciones.

- Naïve Bayes

Es un clasificador probabilístico [53] que determina la categoría de un documento utilizando la probabilidad de un documento $d_i = \{w_{i1}, w_{i2}, \dots\dots, w_{ik}, \dots\dots, w_{iv}\}$ perteneciente a la categoría c_j, $P(c_j|d_j)$. Para determinar esta probabilidad, se aplica el teorema de Bayes.

$$P(c_j|d_i) = \frac{P(c_j)P(d_i|d_j)}{P(d_i)}$$

Donde $P(d_i)$ representa la probabilidad de que un documento escogido al azar tenga un vector di como su representación y $P(c_j)$ es la probabilidad de que un documento escogido al azar pertenezca a la categoría.

- Red de Bayes

La suposición principal del clasificador *Naïve Bayes* (NB) es la independencia de las características. La otra suposición extrema es suponer que todas las características son totalmente dependientes. Esto lleva al modelo de *Red Bayesian* (RB), que es un gráfico acíclico dirigido cuyos nodos constituyen variables aleatorias, y los bordes representan dependencias condicionales. RB se considera un modelo completo para las variables y sus relaciones.

- Máxima Entropía

El clasificador de Máxima Entropía no toma suposiciones con respecto a la relación entre las características. Este clasificador siempre intenta maximizar la entropía del sistema mediante la estimación de la distribución condicional hacia la etiqueta de la clase. La distribución condicional está definida de la siguiente forma:

$$P_\lambda(y|X) = 1/Z(X)exp\left\{\sum_i \lambda_i f_i(X,y)\right\}$$

X es el vector de función e y es la etiqueta de clase. $Z(X)$ es el factor de normalización y λ_i es el coeficiente de peso. $f_i(X,y)$ es la función característica que se define como:

$$f_i(X,y) = \begin{cases} 1, & X = x_i \ y \ Y = Y_i \\ 0, & caso\ contrario \end{cases}$$

- *No Supervisado*

En el aprendizaje supervisado, el objetivo es aprender un mapeo desde la entrada hasta una salida cuyos valores correctos son proporcionados por un supervisor. En el aprendizaje no supervisado, no se presenta tal supervisor y solo contamos con datos de entrada. El objetivo es encontrar las regularidades en la entrada. Hay una estructura en el espacio de entrada de forma tal que ciertos patrones ocurren más a menudo que otros, y queremos ver qué sucede generalmente y qué no.

Léxico

Las palabras de opinión en un tweet pueden ser positivas, esto para expresar algunos estados deseados, mientras que las palabras de opinión negativa se usan para expresar algunos estados no deseados. También hay frases de opinión y expresiones idiomáticas que juntas se llaman léxico de opinión. Hay tres enfoques principales para compilar o recopilar la lista de palabras de opinión. El enfoque manual consume mucho tiempo y no se usa solo. Por lo general, se combina con los otros dos enfoques automatizados como una verificación final para evitar los errores que resultan de los métodos automatizados. Los dos enfoques automatizados se presentan en las siguientes subsecciones:

- Basado en Diccionario

El diccionario es un pequeño conjunto de palabras de opinión que se recopila manualmente con orientaciones conocidas o vinculadas. Luego, este conjunto crece al buscar en los conocidos corpus *WordNet* [54] o tesauro [55] sus sinónimos y antónimos. Las palabras recién encontradas se añaden a la lista de semillas y luego comienza la siguiente iteración. El proceso iterativo se detiene cuando no se encuentran palabras nuevas. Una vez que se completa el proceso, es posible realizar una inspección para descartar o corregir los errores. Este enfoque tiene una gran desventaja, la cual es de encontrar palabras de opinión orientadas o relacionadas al contexto de análisis de sentimientos.

- Basado en Corpus

Este método basado en Corpus ayuda a resolver el problema de encontrar palabras de opinión con orientaciones o relaciones específicas dentro del contexto de análisis. Sus métodos dependen de patrones sintácticos o patrones que ocurren junto con una lista semilla de palabras de opinión para encontrar otras palabras de opinión en un gran corpus.

  - Semántico

Este método proporciona valores de opinión directamente y se basa en diferentes principios para calcular la similitud entre las palabras. Además, este principio da valores de sentimiento similares para palabras semánticamente cercanas. *WordNet*, por ejemplo, ofrece una variedad de relaciones semánticas entre palabras usadas para calcular las polaridades del sentimiento. Considerando esto *WordNet* podría usarse también para obtener una lista de palabras de sentimiento al expandir de forma iterativa el conjunto inicial con sinónimos y

antónimos de las palabras y luego determinar la polaridad del sentimiento para una palabra que se desconozca.

- Estadístico

Este método permite encontrar patrones de ocurrencia de palabras de opinión. Esto podría hacerse derivando de las polaridades de los adjetivos en un corpus. También la polaridad de una palabra puede identificarse mediante el estudio de la frecuencia de palabras en un corpus de texto. Si la palabra aparece con mayor frecuencia en texto positivos, entonces la polaridad es positiva, caso contrario la polaridad seria negativa.

2.3.5. Ensemble

Muchos investigadores se han centrado en el uso de clasificadores tradicionales, como Naïve Bayes, Maximum Entropy, Support Vector Machines, Decision Tree, Random Forest, Neural Network, KNN, etc. Para resolver tales problemas de clasificación. Adicional a esta implementación tradicional, existe también la combinación de múltiples clasificadores para generar un solo clasificador [56]–[58].

Los métodos de Ensemble entrenan a múltiples clasificadores para resolver el mismo problema [57]. En contraste con los enfoques clásicos de aprendizaje, que construyen un clasificador a partir de los datos de entrenamiento, los métodos de conjunto construyen un conjunto de clasificadores y los combinan [59], para luego resolver el problema.

En la fase de clasificación de tweets, se tiene muchas técnicas las cuales pueden ser implementadas en forma individual o en conjunto. Considerando esto para el presente trabajo se va realizar la combinación de técnicas enfocadas en *machine learning* y en léxico, con el motivo de tener una variedad de clasificadores heterogéneos que puedan resolver de manera conjunta un problema. Para saber o conocer el desempeño individual o en conjunto de los clasificadores; a continuación se abordará en el tema evaluación.

2.4. Evaluación

Es muy común que después de realizar la clasificación de texto o de comentarios, se realice la evaluación del método implantado, esto con el motivo de conocer que tan efectivo fue o no el método que se usó para la clasificación de texto. Por tanto, en esta sección se mencionará las métricas más usadas en la evaluación de métodos de clasificación de texto.

Antes de realizar la respectiva evaluación del método de clasificación de texto, es necesario la matriz de contingencia o matriz de confusión, como se muestra en la *Tabla 7*. Cuando un ejemplo es positivo (Clase positiva) puede asignarse una etiqueta positiva o negativa del clasificador, que da como resultado un Verdadero Positivo (a) o un Falso Negativo (c), respectivamente. Cuando un ejemplo es negativo (Clase negativa), el clasificador también puede asignarle una etiqueta positiva o negativa, lo que resulta en un falso positivo (b) o verdadero negativo (d). Se han definido varias medidas basadas en este matriz de confusión.

	Clase Positiva	Clase Negativa
Asignación Positiva	a	b
	(VP – Verdadero Positivo)	(FP – Falso Positivo)
Asignación Negativa	c	d
	(FN –Falso Negativo)	(VN - Verdadero Negativo)

Tabla 7. Matriz de confusión para un clasificador binario.

Las métricas más usadas para determinar el rendimiento de un método de clasificación de texto son las que se detallan a continuación:

- Exactitud: se conoce en el idioma inglés como '*Accuracy*'. Esta medida indica cuan bien el clasificador puede pronosticar correctamente la clase correspondiente del conjunto de datos usados como Test

$$ACC = \frac{VP + VN}{VP + FP + FN + VN}$$

- Precision: indica que tan buenas son las clases o categorías propuestas para el sistema. Para el presenta caso se tendrá 'Positivo' o 'Negativo'.

$$P = \frac{VP}{VP + FP}$$

- Recall: indica que tan bueno es el sistema averiguando las clases o categorías deseadas.

$$R = \frac{VP}{VP + FN}$$

- F1: esta medida relaciona las dos variables que anteriormente se mencionó, tales como *recall* y *precision* para medir el comportamiento general del sistema. Se define como:

$$F1 = 2 * \frac{Precision * recall}{Precision + recall}$$

Para la evaluación de clasificadores, tenemos muchas métricas como por ejemplo: *accuracy*, *recall*, *precision*, F1,etc, las cuales son muy usadas en varias investigaciones [60]–[62]. En el presente trabajo se han considerado el uso de *recall, precision* y F1, ya que cada una de ellas nos indica el desempeño que tiene el sistema referente al clasificador, esto independientemente si el *dataset* se encuentra balaceado o no, esto no ocurre con la métrica *accuracy*, la cual se ve afectada si el *dataset* no es balanceado. Por este motivo no se ha considerado la métrica *accuracy* para el trabajo.

2.5. Aplicaciones de Análisis de Sentimientos

En esta sección se presenta la información acerca de las herramientas, que permiten realizar el análisis de sentimientos en redes sociales, especialmente Twitter. Generalmente la mayoría de estas herramientas utiliza la clasificación positiva, negativa y neutra para predecir el sentimiento de los tweets; a continuación, se detallan las siguientes herramientas: Sentiment viz, Linguakit, Opinion crawl y Socialmention.

- Sentiment Viz[5] es una aplicación web desarrollada por Christopher Healey y Siddarth Ramaswamy, en la en la que se permite estimar y visualizar el sentimiento sobre textos cortos e incompletos. Según los creadores de la aplicación, el sentimiento se define como "una actitud, pensamiento o la sentencia motivada por el sentimiento".

[5] http://www.csc.ncsu.edu/faculty/healey/tweet_viz/

El objetivo de esta herramienta es dar una visualización que presente propiedades emocionales básicas incorporadas en el texto, junto con una medida de la confianza en nuestras estimaciones. A continuación, se muestra la *Figura 11* con el funcionamiento de la herramienta Sentiment Viz.

Figura 11. Herramienta Sentiment Viz

- Linguakit[6] es un portal web, que contiene con un kit de herramientas lingüísticas y de extracción textual, con la finalidad de que cualquier persona interesada puede explorar, analizar y obtener una mejor información acerca de un texto especifico. Adicionalmente Linguakit tiene algunas herramientas como el Analizador de Sentimientos, o Analizador de Opinión, esta herramienta es una de las más usadas y se encuentra disponible en inglés, español, portugués y gallego. El analizador de sentimientos presenta estadísticas de texto en la que puede ser positivo, negativo o neutro. A continuación, se muestra la *Figura 12* con el funcionamiento de la herramienta Linguakit.

[6] https://linguakit.com/es/analisis-completo

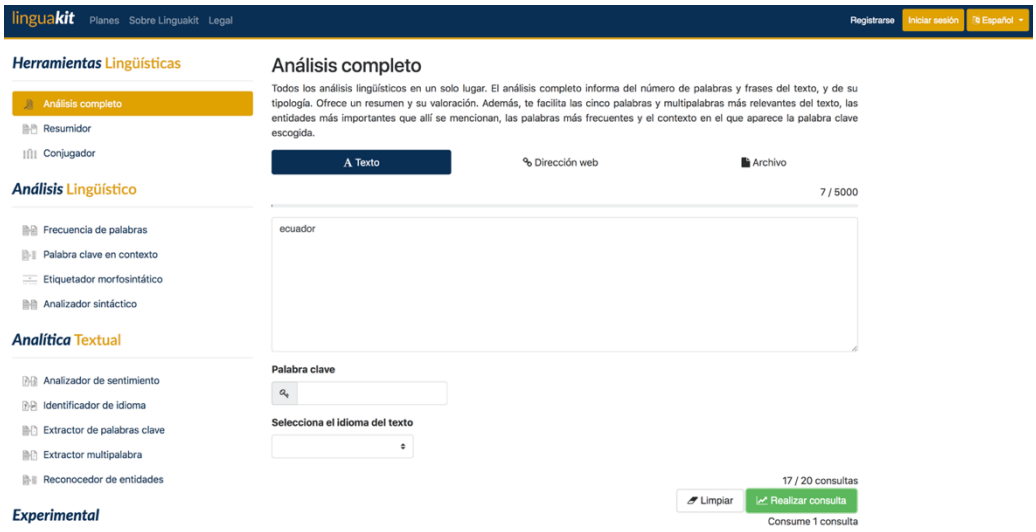

Figura 12. Herramienta Linguakit

- Socialmention[7] es una plataforma web, la cual permite la búsqueda y análisis dentro de medios sociales, esta plataforma no se limita únicamente a Twitter, se puede cambiar la opción de medio social a consultar; la información consultada por el usuario se presenta en tiempo real. El funcionamiento del aplicativo se maneja por 4 parámetros tales como:

  Fuerza: es la probabilidad concerniente a un tema que está siendo mencionando en los medios sociales; y es calculada mediante las menciones dadas en las últimas 24 horas sobre el total de las posibles menciones.

  Sentimiento: es la proporción que existe entre las menciones positivas y negativas.

  Pasión: Medida de probabilidad de que los autores estén mencionando el tema en varias ocasiones.

  Alcance: Medida del rango de influencia que tiene un tema o una palabra dentro de los medios sociales. A continuación, se muestra la *Figura 13* con el funcionamiento de la herramienta Socialmention.

[7] http://socialmention.com

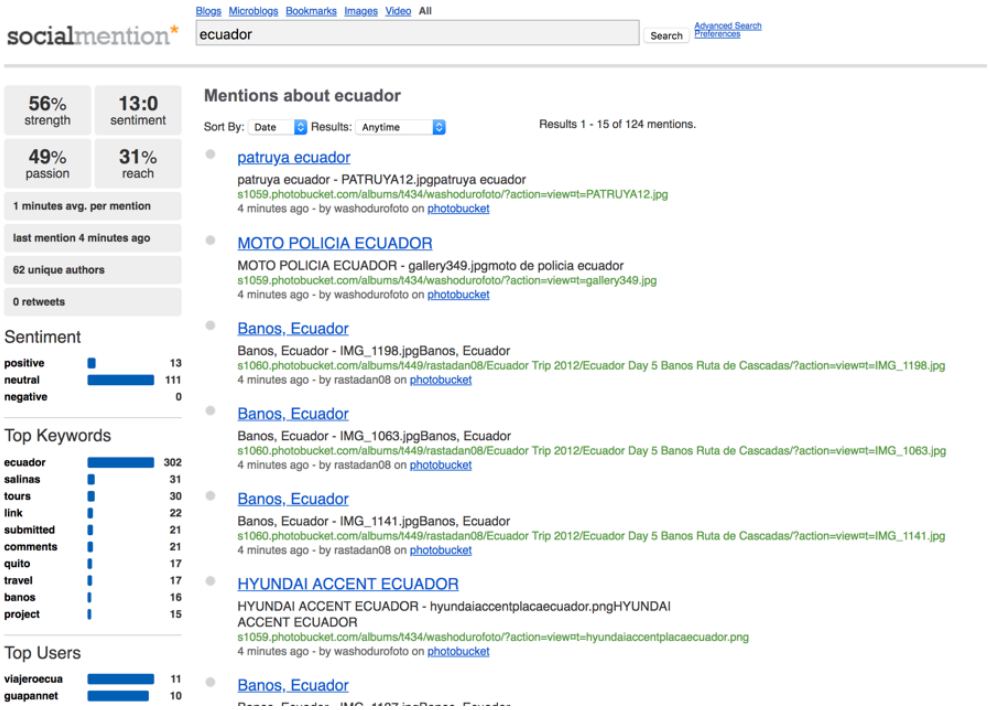

Figura 13.Herramienta Socialmention

- Opinion Crawl[8] es un sitio que permite a los usuarios evaluar el sentimiento de la Web sobre un tema: una persona, un evento, una empresa o un producto; por ejemplo, puede evaluar el sentimiento de un político (Barack Obama), una celebridad (Sandra Bullock), una compañía (Goldman Sachs), un evento actual (derrame de petróleo en el Golfo de México), etc. El visitante puede ingresar un tema y obtener una evaluación de opinión; para cada tema, se obtiene un gráfico circular que expresa el sentimiento actual en tiempo real, una lista de los últimos titulares de las noticias, algunas imágenes en miniatura recientes y una nube de etiquetas de conceptos semánticos clave, que el público asocia con el tema. Los conceptos le permiten ver qué problemas o eventos impulsan el sentimiento de una manera positiva o negativa. A continuación, se muestra la *Figura 14* con el funcionamiento de la herramienta Opinion Crawl.

[8] http://www.opinioncrawl.com

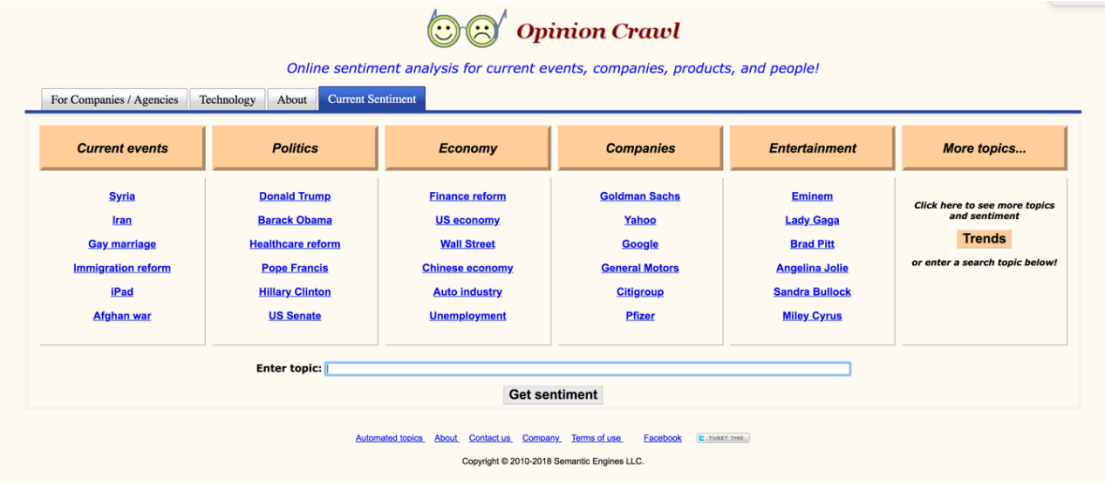

Figura 14. Herramienta Opinion Crawl

Normalmente, dentro de la clasificación de análisis de sentimiento, hay dos enfoques principales de investigación, así como se detalló en las secciones anteriores. Primero tenemos el enfoque léxico, el cual se centra en el uso de diccionarios, y tenemos también al enfoque *machine learning*, el cual está relacionado con vectores de características. Es claro que en muchas investigaciones son mencionadas así como en [21], donde ellos destacan a los métodos supervisados sobre los métodos no supervisados como es léxico. Aunque se debe considerar que los métodos de aprendizaje supervisados demandan gran cantidad de datos o de un corpus para realizar una clasificación adecuada. Dentro de los métodos de aprendizaje supervisado en esta investigación se detalla que SVM presenta una gran precisión a diferencia de otros métodos, pero también tiene limitaciones. También en otra investigación [63] tenemos la combinación de léxico con métodos de *machine learning* , así de esta forma ellos mencionan un gran rendimiento en temas de F1 score y *accuracy*, ya que hacen uso de SVM como método de clasificación y de léxico SentiWordNet, además de modelos matemáticos para la fuerza de sentimiento: *Information Gain* y *Cosine*. En la investigación [25] se menciona la comparación entre métodos supervisados como no supervisados , en los resultados obtenidos por ellos se indica que tanto SVM y Naïve Bayes son muy precisos, a diferencia de métodos léxicos que no son muy efectivos. Dentro de esta investigación se destaca el uso de bigramas como un método para tener un mejor rendimiento durante la clasificación de texto.

En [53] se realizó el respectivo análisis de sentimientos con Twitter mediante la comparación de varias técnicas de preprocesamiento de texto, además de probar con varios métodos de m*achine learning* como SVM, Bernoulli Naïve Bayes y Regresión Logística, ellos

mencionan en los resultados de la investigación que las técnicas de preprocesamiento no recomendadas son: eliminar signos de puntuación, manejo de palabras mayúsculas, reemplazo de palabras alargadas, reemplazo de negaciones y corrección de ortografía, en tanto las técnicas recomendadas por ellos fueron: *stemming*, el reemplazo de repeticiones de puntuación y la eliminación de números. Adicionalmente indican que, dependiendo del clasificador, los resultados varían y si los combinan puede también tener resultados cambiantes. Las pruebas fueron realizadas con dos *dataset*: uno fue Semval 2015 y el otro que SS-Twitter, los cuales se detallan en la sección 2.3.1 - Tabla 2.

En otra investigación [64] se menciona la detección de emociones a nivel de oraciones, ellos implementan dos tipos de léxicos como el estático y un léxico emocional adaptativo al contexto en el que se analiza, este léxico hace uso de otros tres léxicos, que fueron creados para el análisis de sentimientos con respecto a la polaridad, intensidad y emoción. Los resultados que ellos presentan son la reducción de características a diferencia de usar N-gramas, además que los Meta-level features ayudan a la detección automática de emociones en el texto consiguiendo una mejora significativa, a diferencia de POS(*Part-of-Speech*) que implicaba mayor complejidad.

En áreas de aplicación para el análisis de sentimientos, mencionamos esta aplicación enfocada [65] en las elecciones generales de Pakistan, donde se agrupa tweets utilizando un sitio web denominado twimemachine[9] y posteriormente son catalogados, en Pro (a favor) y Anti (en contra) para cada uno de los partidos políticos. Luego ellos hacen uso de técnicas de *machine learning* como: *Decision Tree, Naïve Bayes* y *Support Vector Machine*, en estos tweets para predecir los resultados de las elecciones generales de 2013 en Pakistán. Todas estas técnicas predijeron que el partido Pakistan Tehreek-e-Insaf (PTI) surgiría como ganador, sin embargo, las elecciones reales fueron ganadas por la Liga Musulmana de Pakistán Nawaz (PMLN), por lo tanto, la técnica no fue exitosa. Las principales razones que llevó a esta predicción equivocada fueron el hecho de que Pakistán es un país en desarrollo y que las muestras tomadas no constituían al total de los votantes.

Dentro de la predicción en la bolsa de valores, en temas de oro, plata y petróleo tenemos esta investigación [66] menciona el uso del análisis de sentimientos para realizar la predicción

[9] http://www.twimemachine.com

de estos productos en el mercado. Dentro de la investigación hacen uso de SentiWordNet para la asignación de peso en las palabras, además como clasificador usan Redes Neuronales para el aprendizaje de correlaciones y posteriormente la predicción en el mercado de valores.

La popularidad que tiene un producto dentro del mercado de consumidores puede ser un tema de análisis de sentimientos, en este caso mencionamos a la investigación [67] la cual presenta un análisis de sentimientos en Twitter para conocer la tendencia de compra de celulares que tienen las personas, en esta investigación se usó un método léxico para la clasificación de tweets en: muy positivo, positivo , neutro , negativo y muy negativo, como herramienta de desarrollo se usó lenguaje R, MongoDb como base de datos y además del API de Twitter para la extracción de tweets, este trabajo se enfoca en la popularidad de 5 marcas de celular como son : Samsung, Nexus, Iphone, Lenovo y Motorola dependiendo como se mencionan se puede concluir que tipo de celular es atractivo para la compra en ciertos sectores de la sociedad.

En la investigación [68] se presenta una temática social, como es el tema de los refugiados sirios en Turquía, esto debido a la guerra civil que hay en Siria. La investigación analizó las menciones o tweets vinculados a la acogida que tienen los refugiados en Turquía, así como los temas relacionados sobre este número de personas extranjeras. Para esto la investigación recopila 2 *dataset* en inglés y turco mediante un paquete R con conexión a Twitter, posteriormente para la categorización de tweets ellos generan un léxico con palabras relacionadas a esta temática, por ejemplo: refugiados sirios, guerra etc. En los resultados obtenidos se presenta una realidad de que muchos turcos no consideran bueno la pertenencia de refugiados en su país.

2.6. Conclusiones

El abordaje concerniente al análisis de sentimientos nos ha llevado a investigar los mecanismos necesarios para este fin. Dentro de los mecanismos ya detallados podemos mencionar al preprocesamiento de texto, es aquí donde destacan muchas técnicas empleadas en varias investigaciones, por esta razón se implementará: eliminación de url, eliminación de *stopwords*, eliminación de username. Para la representación de tweets se propone usar la técnica TFID, ya que muchos paquetes incluyen esta representación. En tema de

clasificación de tweets se usará “Ensemble”, donde se combinará varias técnicas de *machine learning* y de léxico. Estas técnicas pueden ser vistas en la sección 2.3.4.

En el siguiente capítulo se presenta la propuesta formal para el desarrollo de este trabajo de disertación, una vez que se ha detallado los conceptos, así como el estado del arte, en relación a las fases o etapas dentro del análisis de sentimientos para la red social Twitter y su respectiva implementación sobre el aplicativo móvil.

3. Análisis de sentimientos

3.1. Introducción

En este capítulo se detallará la propuesta realizada para el desarrollo del presente trabajo de disertación, en donde se abordarán los mecanismos que consideramos necesarios para el análisis de sentimientos en Twitter. Adicionalmente se mencionarán los requerimientos funcionales, no funcionales, arquitectura del sistema y funcionamiento de la solución.

En el presente trabajo se propone realizar un análisis de sentimientos sobre la red social Twitter, para esto se extraerán tweets acerca de temas relevantes y sobre algún tema de interés particular, planteado por el usuario final. Una vez extraídos los tweets se realizará el procesamiento de texto, donde se implementará las técnicas de limpieza de texto y selección de características. Posterior a este paso se realizará la clasificación de tweets mediante Ensemble, la cual consiste en agrupar varias técnicas, como puede ser de *machine learning* o léxico dentro del proceso de clasificación. Cada una de las técnicas empleadas, entrarán en un proceso de evaluación para determinar su rendimiento y así presentar finalmente la información, mediante un aplicativo móvil. En la *Figura 15* se muestra la arquitectura general propuesta para el análisis de sentimientos en Twitter.

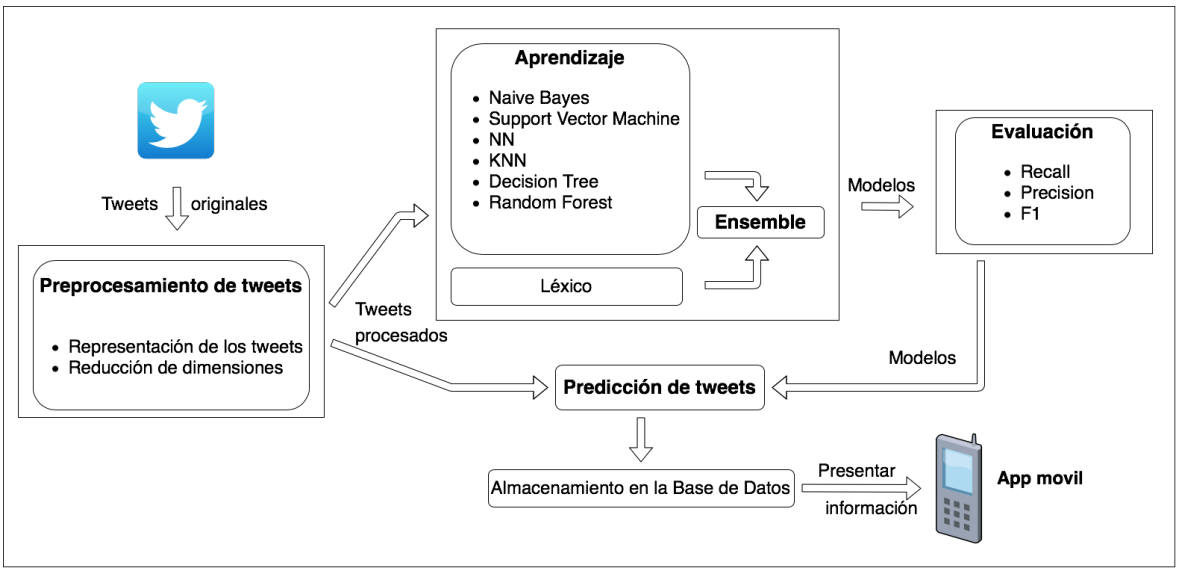

Figura 15. Arquitectura general de la propuesta

Dentro de la arquitectura podemos observar algunos componentes tales como: preprocesamiento de tweets, aprendizaje, evaluación, predicción de tweets, almacenamiento en la base de datos y como último componente el aplicativo móvil. Cada uno de estos componentes se han dividido en dos propuestas: la primera propuesta de análisis de sentimientos y la segunda como propuesta de aplicativo móvil.

3.2. Análisis de sentimientos

Como propuesta para el análisis de sentimientos en Twitter, se ha considerado las fases presentadas en el capítulo anterior, las cuales son: la adquisición de datos, el preprocesamiento de tweets, la clasificación de tweets y finalmente la evaluación. En las secciones posteriores se detallarán cada uno de estos componentes.

3.2.1. Adquisición de datos

Para la adquisición de datos se propone extraer tweets referentes a una temática en particular, para posteriormente realizar una clasificación manual ("positivo o negativo"). Este *dataset* previamente clasificado, nos permitirá realizar el entrenamiento de cada uno de los clasificadores, así como predecir y medir el rendimiento de cada uno de ellos. Adicional a este *dataset* se efectuará la extracción de los tweets más recientes, lo cual permitirá la predicción y generación de un nuevo *dataset* que incluya estos tweets ya clasificados.

3.2.2. Preprocesamiento de tweets

En procesamiento de texto es un paso muy importante para el análisis de sentimientos, ya que permite la reducción de ruido, dimensionalidad y selección de características significantes para el análisis de sentimientos. Para esto se ha considerado las técnicas de preprocesamiento texto mencionadas en la sección 2.3.2 y que son usadas en varias en investigaciones. Para esto se ha dividido en: limpieza de tweets y selección de características; a continuación, son presentadas cada una de las técnicas:

- Para la limpieza de tweets:
    - Eliminación de URLs.
    - Eliminación de *stopwords*.
    - Eliminación de hashtags.
    - Eliminación de espacios en blanco.
    - Eliminación de signos de puntuación.
    - *Lowercase*.

- Selección de Características
    - TF-ID

Una vez realizado el preprocesamiento de texto sobre el *dataset* de entrenamiento y de test, procedemos con la clasificación de tweets. Para esto, en la siguiente sección se dará un mayor detalle al respecto.

3.2.3. Clasificación de tweets

Para la clasificación de tweets en positivos y negativos, se propone el uso de una técnica meta la cual combina varias técnicas de machine de *machine learning* o de léxico. A continuación, se enumeran las técnicas empleadas para la propuesta. Para la selección de estas técnicas, se consideró técnicas heterogéneas y las más comunes, dentro de los trabajos de investigación presentados en la sección 2.3.4.

- Machine learning
    - Naïve Bayes
    - SVM
    - Decision Tree
    - Random Forest
    - Neural Network
    - K neighbour nearest
- Estadística
    - Léxico

Dentro del “Ensemble” se ha tomado algunas variantes, para la realización de pruebas, por esta razón se tiene “Ensemble top 5” y “Ensemble top 4”. Estas variantes de “Ensemble” consisten en tomar las mejores técnicas puntuadas por la métrica F1 de forma individual. En el caso del “Ensemble top 5” se tomará las 5 mejores técnicas puntuadas por F1 y “Ensemble top 4” se tomará las 4 mejores puntuadas por F1; a continuación, tenemos la *Tabla 8*, la cual presenta el mecanismo en que el “Ensemble” realiza la clasificación por votación mayoritaria:

Técnicas de Clasificación							**Ensemble**
NV	SVM	NN	RF	KN	DT	Léxico	
Positivo	Positivo	Positivo	Positivo	Positivo	Negativo	Negativo	Positivo

Tabla 8. Clasificación de tweets mediante “Ensemble”

Una vez realizada la clasificación de polaridad de cada uno de los tweets, se procede con la evaluación de rendimiento; para esto, en la siguiente sección se dará un mayor detalle al respecto.

3.2.4. Evaluación

Para el componente de evaluación, nosotros hemos considerado algunas métricas para determinar el rendimiento de cada una de las técnicas empleadas, en la clasificación de tweets. A continuación, se enumera las métricas usadas sobre los clasificadores binarios, (para mayor detalle ver la sección 2.4).

- Precision
- Recall
- F1

3.3. Aplicación móvil

Como propuesta de desarrollo, se plantea desarrollar un aplicativo móvil, el cual permita realizar un análisis de sentimientos en la red social Twitter, ya sea por temas relevantes o por una consulta solicitada por el usuario final. El análisis de sentimientos mostrado en el aplicativo móvil consistirá en la polaridad (“positiva o negativa”) que tenga ese tema dentro

de la red social. Esta polaridad presentada brindara información al usuario, para que él pueda tomar la mejor decisión frente alguna problemática.

Cabe destacar que la construcción del aplicativo móvil comenzó desde un Paper "*Harvesting Opinions in Twitter for Sentiment Analysis*" previamente elaborado, donde se propuso realizar un análisis de sentimientos con la ayuda de un método estadístico, específicamente "léxico". Teniendo en cuenta este antecedente, se propone un aplicativo móvil en la plataforma iOS, dando la posibilidad al usuario de consultar y obtener una información de manera rápida y confiable.

A continuación, se presentará el prototipo del aplicativo móvil, la arquitectura del sistema, y el funcionamiento de la solución.

3.3.1. Prototipo

Para la realización del prototipo se ha divido en dos módulos como son: online y offline. A continuación, se presentarán las imágenes con la funcionalidad que cada una de ellas desempeña en el aplicativo móvil.

- Módulo online

En el modulo online, el usuario final podrá realizar la consulta en tiempo real sobre cual temática y podrá obtener la polaridad ("positiva o negativa"). Para esto se ha considerado los siguientes componentes gráficos como se visualiza en la *Figura 16*.

  - Caja de texto: donde el usuario ingresara la temática hacer consultada.
  - Botón de buscar: componente para dar inicio a la busque de la palabra ingresada.
  - Grafico de pastel: presentará el porcentaje de polaridad positiva o negativo, a ese tema.
  - Mensaje de error: se mostrará cuando el servicio no este disponible.

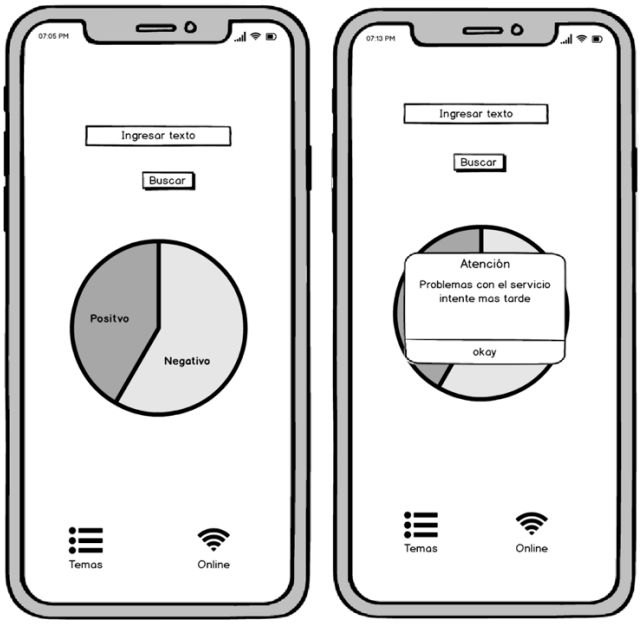

Figura 16. Prototipo módulo online

- Módulo offline "Temas"

En el módulo offline, el usuario final podrá seleccionar 4 temas relevantes, tales como: "Ecuador", "Lenin Moreno", "Islas Galápagos" y "Correa" así como se muestra en la Figura *17*.

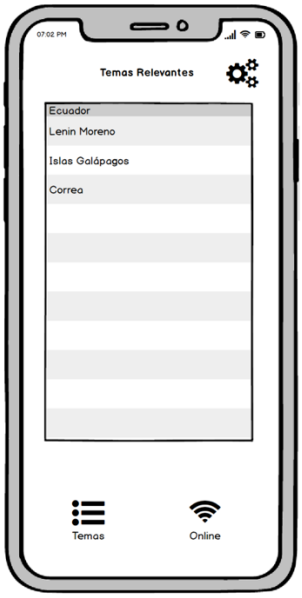

Figura 17. Prototipo módulo offline (parte 1)

Una vez seleccionado el tema se presentará un gráfico de barras indicando la polaridad ("positiva o negativa") así como se muestra en la *Figura 18*, en esta pantalla se puede presentar un mensaje de error si el servicio no está disponible en ese momento.

Figura 18. Prototipo modulo offline (parte 2)

- Módulo online

En el módulo online el usuario final puede modificar el intervalo de días para la consulta offline, así como se muestra en la Figura 19.

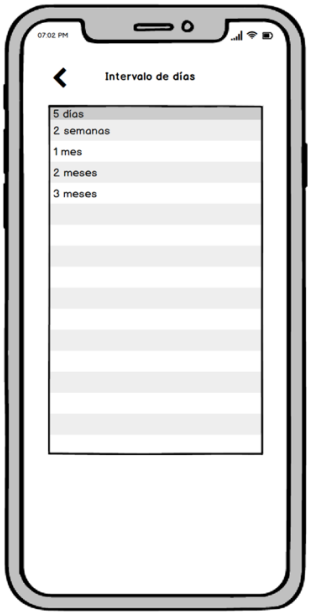

Figura 19. Prototipo módulo offline (parte 3)

El prototipo que se propone desarrollar forma parte de un de conjunto de componentes llamado arquitectura. En la siguiente sección se presentará con mayor detalla la arquitectura del sistema, que se pretende realizar.

3.3.2. Arquitectura del Sistema

Para el desarrollo del sistema propuesto se utilizará el modelo cliente/servidor en tres capas. La arquitectura de diseño de software cliente/servidor en tres capas, se define como una organización jerárquica tal que cada capa proporciona servicios a la capa inmediatamente superior y se sirve de las prestaciones que le brinda la inmediatamente inferior [69]. Por esta razón se dividirá los componentes de la aplicación en tres niveles tal como se muestra a continuación en la *Figura 20*:

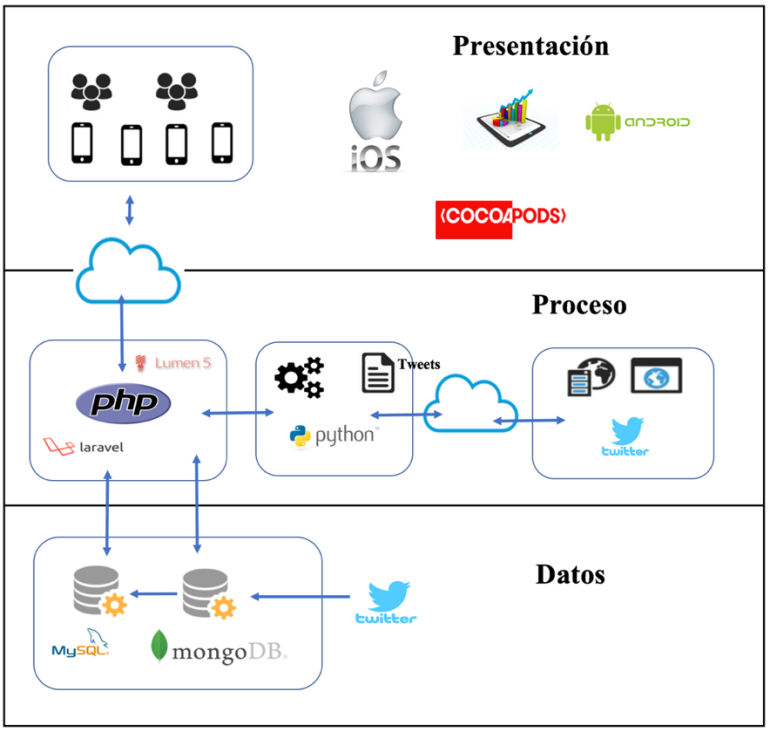

Figura 20. Arquitectural general del sistema

- **Capa de presentación**: Hace referencia a la interfaz gráfica que se encarga de recoger la información al usuario y enviarla al servidor, para posteriormente ser procesada y presentar un resultado final al usuario.
- **Capa de proceso**: se encarga de recibir la entrada de datos de la capa de presentación e interactuar con la capa de datos para realizar operaciones, guardar información o

extraerla y procesar los resultados para posteriormente mostrarlos al usuario a través de la capa de presentación.

- **Capa de datos:** Almacena los datos, los recupera, mantiene y asegura su integridad, interactúa únicamente con la capa de proceso.

3.3.3. Funcionamiento de la solución

Para poder visualizar, especificar y complementar el diseño del prototipo de software, el cual permite realizar el análisis de sentimientos en español, a través de un motor de bases de datos no relacionados, se hace uso de algunos diagramas del lenguaje gráfico para el modelado de sistemas de software, UML.

A continuación, se muestran los diagramas que se creyeron necesarios para poder representar de forma ágil y entendible el sistema:

- ***Diagramas de Flujo***

Los diagramas de actividades permiten ver el proceso de software como un flujo de trabajo de inicio a fin, en ellos se muestra la secuencia de actividades que son necesarias para cumplir con una tarea, junto con las decisiones que se presentan durante la ejecución de los eventos de la actividad.

En la *Figura 21* se muestra el diagrama de actividades, referente a la consulta online con la que el usuario final puede interactuar con el análisis de tweets, en este diagrama las acciones y decisiones en color gris representan las que son realizadas por el usuario final de la herramienta, y las que no tienen ningún color son las creadas por el sistema.

- La actividad inicia cuando el usuario final ingresa la palabra hacer analizada.
- Luego se realiza la petición al *webservice*, en donde efectua la extracción de 50 tweets con la ayuda del API de Twitter.
- Una vez extraídos los tweets, se genera un *dataset* que pasa a la fase de preprocesamiento de texto, en esta fase se tiene las técnicas tales como: *lowercasing*, remover *hyperlinks*, remover *stopworks*, remover hashtags y remover signos de puntuación.
- Posterior al paso de preprocesamiento de texto se consulta al *dataset* para saber si la palabra ingresada por el usuario existe o no, en el *dataset*.

- Si la palabra ingresada por el usuario es nueva dentro del *dataset*, se procederá con la clasificación de los tweets descargados. El proceso de clasificación de tweets en positivos o negativos, se realiza con la técnica estadística léxico, esta técnica utiliza un diccionario que contiene 4276 palabras clasificadas entre positivas y negativas.
- Después de ser clasificados cada uno de los tweets, se realiza el almacenamiento en el *database* para una posterior consulta o análisis del usuario.
- Si la palabra ingresada por el usuario ya fue consultada previamente o existe en el *dataset*, se extraerán los tweets del *database* para obtener el *dataset* referente a esa palabra.
- Con el *dataset* procedente del *database*, se procederá con la clasificación de los tweets, para esto se usará la técnica meta, la cual combina técnicas de *machine learning* y de léxico para la clasificación de los tweets.
- Después de ser clasificados cada uno de los tweets, se realiza el almacenamiento en el *database* para una posterior consulta o análisis del usuario.
- Como última actividad se tiene la presentación de los datos en el aplicativo móvil, para esto se solicita y devuelven los datos por parte del *webservice* para que finalmente la información sea desplegada en la capa de presentación.

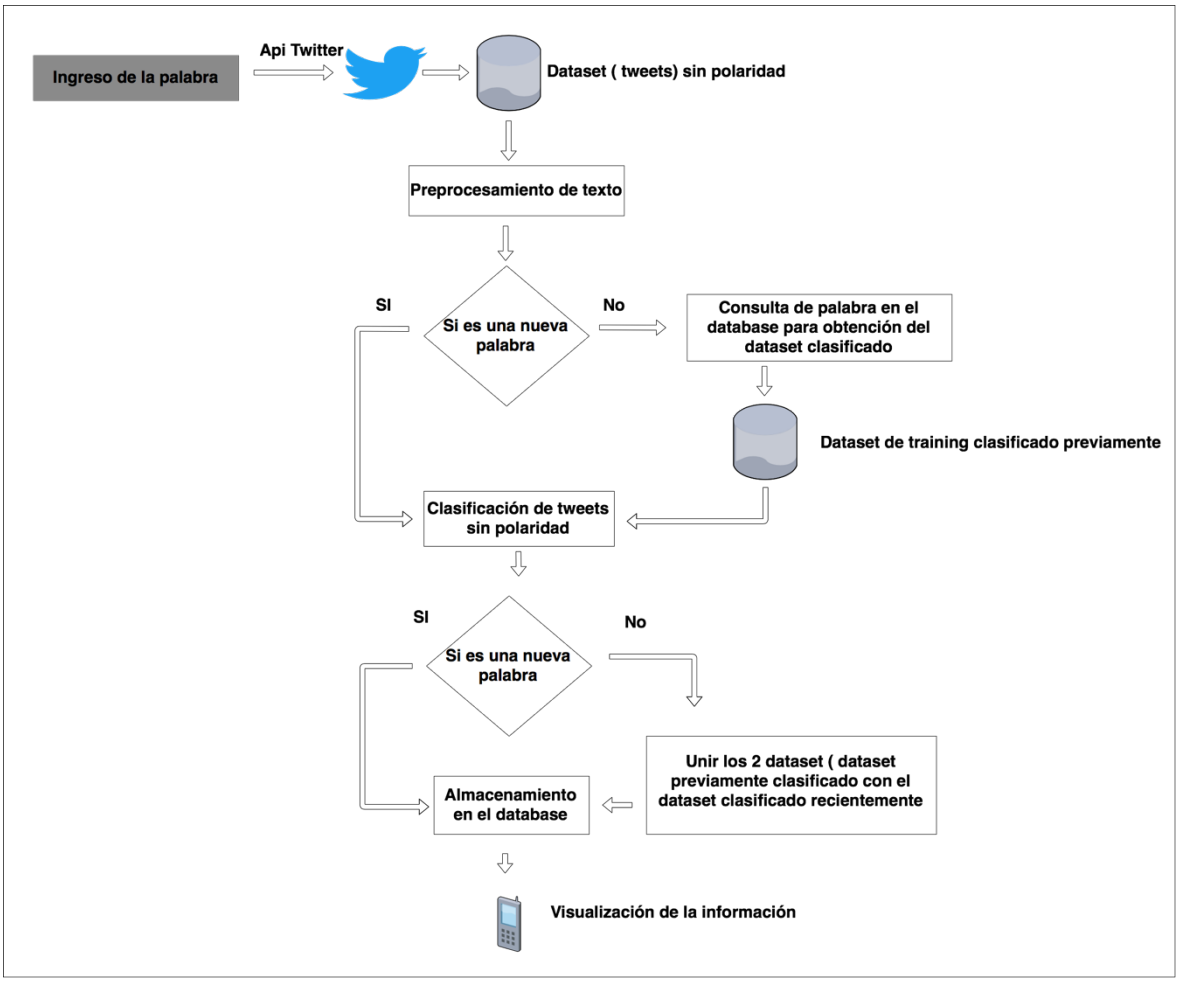

Figura 21. Diagrama de actividades para una consulta online

En la *Figura 22* se muestra el diagrama de actividades, para realizar la consulta referente a temas predefinidos. Estos temas puedes ser analizados por el usuario final mediante el aplicativo móvil.

- Las actividades de la *Figura 22* son semejantes a las de la *Figura 21*, varían en tema de ejecución ya que las actividades de la *Figura 22* se ejecutan todos los días, con la finalidad de obtener información clasificada y disponible al usuario final, además de esta funcionalidad, se añade una nueva actividad denominada suma de datos. Esta actividad consiste en realizar una búsqueda en la base de datos por un intervalo de tiempo, sobre el tema seleccionado por el usuario, para posteriormente presentar la información en el aplicativo móvil, mediante el uso de un gráfico estadístico.

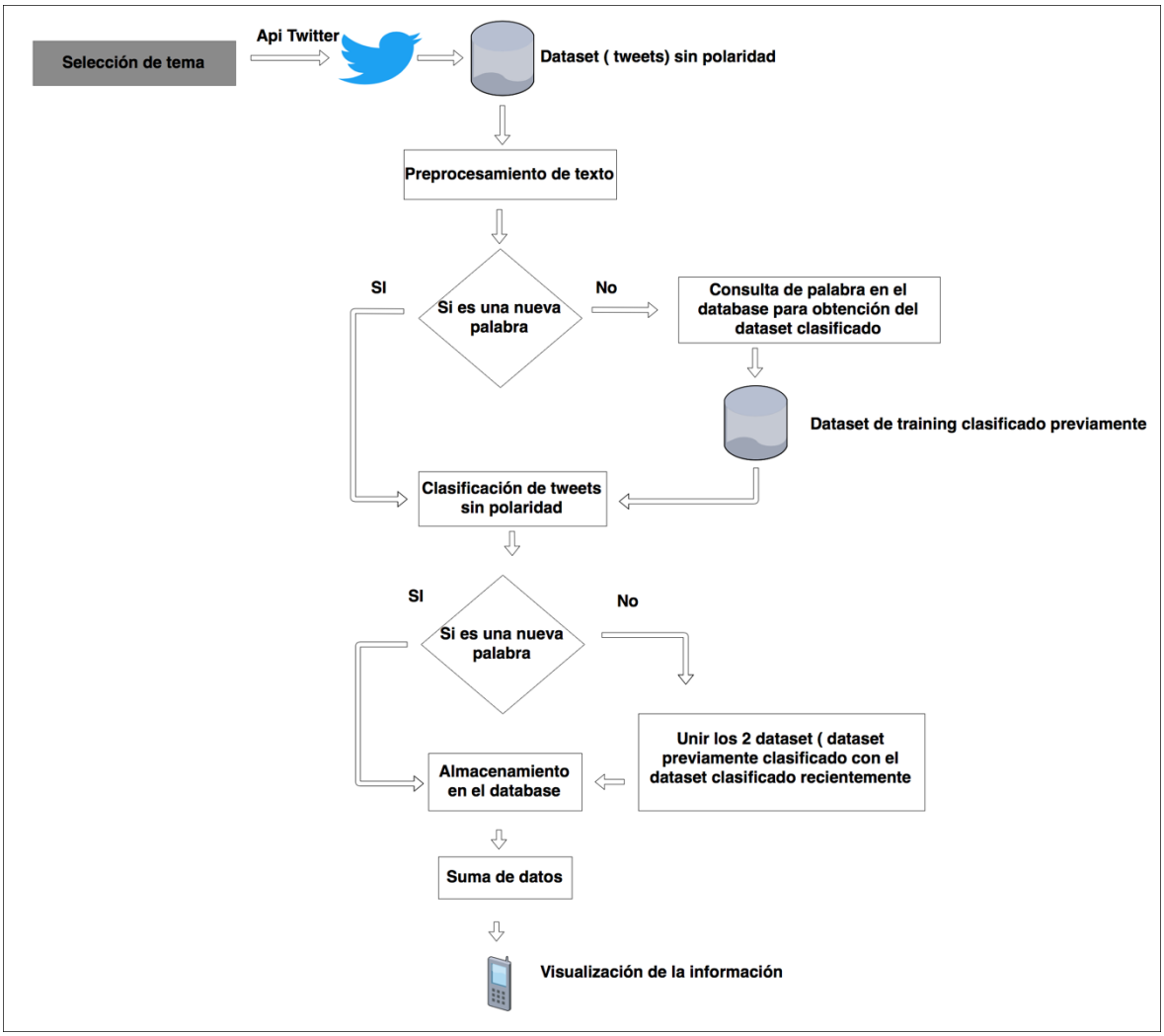

Figura 22. Diagrama de actividades para una consulta offline.

3.3.4. Requerimientos

En sección se detallará cada uno de los requerimientos del aplicativo móvil, para esto se ha dividido en 2 módulos: requerimientos del aplicativo móvil y requerimientos del *webservice*.

- ***Requerimientos del aplicativo móvil***

Requerimientos Funcionales

- El aplicativo móvil contará con una barra de opción, donde el usuario puede acceder a los tipos de consulta (online y temas relevantes), adicional se tiene otra opción de configuraciones donde se muestra el rango de fecha para realizar la consulta de los datos.
- El aplicativo móvil presentará la polaridad positiva o negativa, que el usuario seleccione, esto referente a una temática.

Requerimientos No Funcionales

- Usabilidad

El aplicativo móvil debe ser de fácil uso y comprensión por parte de los usuarios, debe mantener áreas definidas para la manipulación de la información, botones, menú y mensajes.

- Disponibilidad

El aplicativo móvil estará en capacidad de dar respuesta al acceso de todos los usuarios y sus procesos.

- ***Requerimientos del webservice***

Requerimientos Funcionales

- Una vez que se extraiga los tweets se realizará el preprocesamiento de texto, clasificación y predicción, para que finalmente presentar la polaridad correspondiente.
- Se clasificará y se almacenará diariamente, nuevos tweets referentes a los temas relevantes.
- Se almacenará en la base de datos los tweets que se consideren aptos durante el análisis de sentimientos.

Requerimiento No Funcionales

- Escalabilidad – Capacidad

El *webservice* deberá garantizar el crecimiento del volumen de la información almacenada que se manejará en la base de datos.

- Confiabilidad

La clasificación de tweets deberá garantizar una métrica de F1 del 70%, esto para aceptar la información final presentada al usuario.

3.4. Conclusiones

La propuesta para el presente trabajo de disertación se ha definido como el desarrollo de un aplicativo móvil en la plataforma iOS, el cual permita a los usuarios realizar un análisis de sentimientos sobre la red social Twitter, de una forma rápida y confiable. Para este objetivo se ha considerado la arquitectura 3 capas bien definidas tales como: datos, presentación y proceso.

En el siguiente capítulo se presenta la implementación requerida, para desarrollar la propuesta contemplada en la disertación, adicionalmente en este capítulo, se presentarán las pruebas realizadas durante el desarrollo del trabajo.

4. Implementación

4.1. Introducción

En este capítulo se explicará la implementación para el análisis de sentimientos en Twitter, en base a la propuesta planteada en el capítulo anterior. Adicionalmente se mostrarán fragmentos de código necesarios para el análisis de sentimientos.

Para la implementación del presente trabajo, fue necesario el uso de ciertas herramientas, tales como lenguajes de programación especializados en el análisis de sentimientos, así como en servicios web y de interfaz gráfica de usuario. Todo este conjunto de herramientas proporciona una pieza clave, durante el desarrollo del sistema, es por esto, que en las siguientes secciones se presentará con mayor detalle cada una ellas.

4.2. Herramientas utilizadas

En esta sección se detallarán cada una de las herramientas usadas para el desarrollo de la propuesta, dentro de ellas tenemos: Python, Laravel, Swift, CocoaPods, Mongodb y Mysql. Cada una de ellas fueron seleccionadas por sus especificaciones o características, vinculadas y necesarias para la realización del sistema de análisis de sentimientos en Twitter; a continuación, se presentarán con mayor detalle las herramientas.

- ***Python***

El lenguaje de programación a utilizar en el desarrollo del Análisis de Sentimientos en Twitter es Python v2.7, el cual maneja una filosofía y hace hincapié en una sintaxis que favorezca a un código legible. Es importante también destacar, que Python es un lenguaje de código abierto, que contiene muchas librerías y paquetes disponibles para la comunidad en general permitiendo así, el desarrollo de aplicaciones especializadas[10].

[10] https://www.python.org

Figura 23. Icono del lenguaje de programación Python

Para el presente trabajo se ha seleccionado al lenguaje de programación Python, ya que cuenta con: paquetes especializados en extracción de tweets, procesamiento de lenguaje natural y técnicas de *machine learning*, requeridas para el desarrollo del sistema propuesto.

- ***Laravel***

El lenguaje de programación a utilizar para el desarrollo del *webservice* es PHP v7, dentro del universo de frameworks se ha elegido Laravel Lumen v5.5, ya que es de código abierto y dispone de una documentación actualizada, además de ser un framework fácil de entender y ligero en tamaño[11].

Figura 24. Icono del framework Laravel

- ***Swift***

El lenguaje de programación destinado para el desarrollo de la interfaz gráfica de usuario es Swift v4, ya que es un lenguaje de programación multiparadigma creado por Apple enfocado en el desarrollo de aplicaciones para iOS y macOS[12].

Figura 25. Icono del lenguaje de programación Swift

[11] https://lumen.laravel.com

[12] https://developer.apple.com/swift/

- ***CocoaPods***

Como gestor de dependencias, se ha seleccionado a CocaPods ya que cuenta con más de 47 mil bibliotecas y se usa en más de 3 millones de aplicaciones. CocoaPods brinda ayuda en temas de escalamiento durante el desarrollo del aplicativo[13].

Figura 26. Icono del gestor de dependencias CocoaPods

- ***MongoDB***

Como gestor de base de datos, se ha seleccionada MongoDB v2.6.10, ya que es un sistema de base de datos NoSQL orientado a documentos, desarrollado bajo el concepto de código abierto[14]. Por lo anteriormente mencionado se usará esta base de datos para almacenar todos los tweets descargados desde el API de Twitter.

Figura 27. Icono de la base de datos Mongodb

- ***MySQL***

Como gestora de base de datos, también se usará MySQL, ya que es una base de datos relacional de código abierto y que cuenta con documentación actualizada[15]. Para el presente trabajo se usará esta base de datos para almacenar datos preliminares de polaridad, que serán mostrados en la capa de presentación.

[13] https://cocoapods.org

[14] https://www.mongodb.com

[15] https://www.mysql.com

Figura 28. Icono de la base de datos MySQL

4.3. Desarrollo del aplicativo

Para la implementación se ha definido un conjunto de pasos, esto con la finalidad de realizar el análisis de sentimientos en Twitter. Teniendo en cuenta lo anteriormente mencionado se seguirá la Figura 29 que se presenta a continuación.

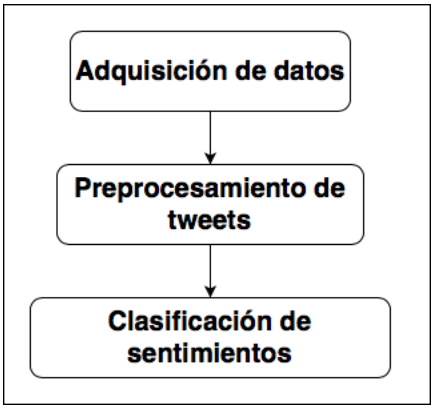

Figura 29. Diagrama de flujo para clasificación de sentimientos

Basándonos en la Figura 29, primeramente, se realizará una descarga masiva de tweets, para luego realizar el preprocesamiento de estos y presentar un método encargado de crear el vector de características fundamentales, esto permitirá entrenar al clasificador con el que posteriormente, se obtendrá la polaridad de los tweets.

4.3.1. Descarga masiva de tweets desde el API de Twitter

En este apartado se hablará de cómo se ha realizado el procedimiento para descargar los tweets de forma masiva desde la API de Twitter. El mecanismo empleado, fue el servicio API REST que cuenta Twitter, mediante el uso de la librería Tweepy de Python, por este

mecanismo se pudo acceder a los tweets que contengan una palabra en común durante la búsqueda.

En la *Figura 30* se menciona dos librerías de Python, necesarias para la extracción masiva de tweets.

- Tweepy = Es un paquete de python, que permite acceder al API de Twitter. Los datos extraídos por el api son: tweet, username, localización y fecha de creación del tweet.
- Pandas = Es un paquete de Python, que permite tener una representación en memoria de cada uno de tweets que son extraídos.

```
269     import tweepy
270     import pandas as pd
```

Figura 30. Código para descarga masiva de tweets.

En la *Figura 31* se detallan las credenciales necesarias para acceder al API de Twitter.

```
277     consumer_key='AtIH3ihXfS6jiC27CzFtMEp5w'
278     consumer_secret='DIi3kQbsBvzwe2HIflcy7xCay3k5YpnbHaKJkcDWj08jtpzUqS'
279     access_token_key='579570553-FrBjlOzCBm34vp9Jyf9JffqSswU9mEMVbvNviEOZ'
280     access_token_secret='y1lK8VcKARxwC8AlIPGXMtbHz9qFl5nXxi3OUvyweArwc'
281     auth=tweepy.OAuthHandler(consumer_key,consumer_secret)
282     auth.set_access_token(access_token_key,access_token_secret)
283     api = tweepy.API(auth)
```

Figura 31. Credenciales para utilizar API de Twitter

En la *Figura 32* se presenta el consumo del API de Twitter, mediante el método *tweepy.Cursor()*. También podemos observar que la extracción de tweets no superara los 50 tweets, además de que los tweets serán en idioma español.

```
284     #Data Result ----
285     data_resul = pd.DataFrame(columns=('word','tweet', 'polaridad', 'user',
286         'location','date','id_categoria','creation_date'))
287     max_tweets = 50
288     try :
289         cont = 0
290         for tweet in tweepy.Cursor(api.search,q=word,result_type="mixed",
291             include_entities=True,lang="es").items(max_tweets):
```

Figura 32. Código para la extracción de tweets

Análisis del Corpus

La idea principal de tener un corpus es para simular la situación más desfavorable para el algoritmo de clasificación. El corpus de "Moreno" del 2017 que hablaremos a continuación,

cuya fiabilidad de estar bien etiquetado servirá para el *dataset test* con el que probaremos los algoritmos de *machine learning*.

Por otra parte, el corpus "Moreno" nos servirá como entrenamiento para los clasificadores, ya que tiene un alto grado de fiabilidad, por haber sido construido de forma manual. El corpus está compuesto por más de 1718 tweets balanceados entre positivos/negativos. La clasificación de cado uno de los tweets, se realizó mediante el criterio de favoritismo, sobre el candidato a la presidencia del Ecuador Lenin Moreno.

```
321         data_train = pd.read_csv('/var/www/html/ASentimientos/Process/leninMorenoPosNeg859.csv',
322             sep = ',', encoding='utf- 8', error_bad_lines=False)
```

Figura 33. Código para carga de dataset "Moreno"

Para el almacenamiento y gestión de tweets, se utilizará la base de datos MongoDB; a continuación, se muestra la conexión con la base de datos y la consulta para determinar si es una palabra nueva:

```
17  from pymongo import MongoClient
18  client = MongoClient()
19  db = client["ASentimientos"]
20  collection = db["tweets"]
```

Figura 34. Conexión con la base de datos mongodb

```
312         if collection.find({"word":word}).count() > 0: # Si ya existe la palabra
```

4.3.2. Configuración para el preprocesamiento de tweets y selección de características

En esta sección se presenta una de las partes más importantes del código, que corresponde a la extracción y limpieza de cada tweet. La sección se encuentra dividida en preprocesamiento de tweets y selección de características.

Para la consulta online dentro del aplicativo móvil, se aplicará las siguientes técnicas de preprocesamiento de tweets:

- Eliminación de URLs.
- Eliminación de hashtags.
- Eliminación de espacios en blanco.

- Eliminación de signos de puntuación.
- Lowercase.

```
def tweet_clean(tweet):
    # Remove tickers
    sent_no_tickers=re.sub(r'\$\w*','',tweet)
    tw_tknzr=TweetTokenizer(strip_handles=True, reduce_len=True)
    temp_tw_list = tw_tknzr.tokenize(sent_no_tickers)
    # Remove hyperlinks
    list_no_hyperlinks=[re.sub(r'https?:\/\/.*\/\w*','',i) for i in temp_tw_list]
    # Remove hashtags
    list_no_hashtags=[re.sub(r'#', '', i) for i in list_no_hyperlinks]
    # Remove Punctuation and split 's, 't, 've with a space for filter
    list_no_punctuation=[re.sub(r'['+string.punctuation+']+', ' ', i) for i in list_no_hashtags]
    # Remove multiple whitespace
    new_sent = ' '.join(list_no_punctuation)
    #Remove any whitespace at the front of the sentence
    clean_sent=new_sent.lstrip(' ')
    return clean_sent
```

Figura 35. Código para preprocesamiento de tweets (Opción online)

Para la consulta predefinida dentro del aplicativo móvil, se aplicará una técnica adicional como es la eliminación de stopwords.

```
def tweet_clean(tweet):
    # Remove tickers
    sent_no_tickers=re.sub(r'\$\w*','',tweet)
    tw_tknzr=TweetTokenizer(strip_handles=True, reduce_len=True)
    temp_tw_list = tw_tknzr.tokenize(sent_no_tickers)
    # Remove stopwords
    list_no_stopwords=[i for i in temp_tw_list if i.lower() not in     cache_spanish_stopwords]
    # Remove hyperlinks
    list_no_hyperlinks=[re.sub(r'https?:\/\/.*\/\w*','',i) for i in list_no_stopwords]
    # Remove hashtags
    list_no_hashtags=[re.sub(r'#', '', i) for i in list_no_hyperlinks]
    # Remove Punctuation and split 's, 't, 've with a space for filter
    list_no_punctuation=[re.sub(r'['+string.punctuation+']+', ' ', i) for i in list_no_hashtags]
    # Remove multiple whitespace
    new_sent = ' '.join(list_no_punctuation)
    #Remove any whitespace at the front of the sentence
    clean_sent=new_sent.lstrip(' ')
    return clean_sent
```

Figura 36. Código para preprocesamiento de tweets (Opción offline)

- ***Configuración para la selección de características***

En esta sección se muestra el código necesario para transformar los tweets como un espacio vectorial, que ayudará posteriormente en el aprendizaje de cada uno de los algoritmos de *machine learning*; para esto usaremos las librerías de: Python NLTK y SKLEARN.

El código presentado a continuación en la *Figura 37* realiza una tokenizacion del tweet, con la finalidad de tener una representación vectorial y posteriormente efectuar *"tf–idf "(Term Frequency times Inverse Document*), que consiste en determinar la frecuencia inversa del término dentro del documento.

```
72      #Extracting features
73      #1) Tokenizing text
74      from sklearn.feature_extraction.text import CountVectorizer
75      count_vect = CountVectorizer()
76      X_train_counts = count_vect.fit_transform(training_texto)
77
78      from sklearn.feature_extraction.text import TfidfTransformer
79      tfidf_transformer = TfidfTransformer()
80      X_train_tfidf = tfidf_transformer.fit_transform(X_train_counts)
```

Figura 37. Código para extracción de características

4.3.3. Configuración para la clasificación de tweets

En esta sección se hablará de como se ha implementado los clasificadores, mediante el uso de la librería SciKit-Learn que es la base del *machine learning* en Python. En la *Figura 38* se presenta el código para el entrenamiento del clasificador, mediante el algoritmo *Naïve Bayes* y posteriormente su predicción.

```
94      # Training a classifier
95      if op ==1: #Naive Bayes
96          from sklearn.naive_bayes import MultinomialNB
97          clf = MultinomialNB().fit(X_train_tfidf, training_polarity)
98          X_new_counts = count_vect.transform(test_texto)
99          X_new_tfidf = tfidf_transformer.transform(X_new_counts)
100
101         predicted = clf.predict(X_new_tfidf)
102         return predicted
```

Figura 38. Código para la clasificación de tweets mediante naïve bayes

En la *Figura 39* se presenta el código para el entrenamiento del clasificador, mediante el algoritmo *Support Vector Machine* (SVM) y posteriormente su predicción.

```
104     if op ==2: # SVM
105         from sklearn import svm
106         clf = svm.LinearSVC()
107         clf.fit(X_train_tfidf, training_polarity)
108         X_new_counts = count_vect.transform(test_texto)
109         X_new_tfidf = tfidf_transformer.transform(X_new_counts)
110
111         predicted = clf.predict(X_new_tfidf)
112         return predicted
```

Figura 39. Código para la clasificación de tweets mediante svm

En la *Figura 40* se presenta el código para el entrenamiento del clasificador, mediante el algoritmo *Decision Tree* y posteriormente su predicción.

```
115     if op ==3:  # Decision Tree
116         from sklearn import tree
117         clf = tree.DecisionTreeClassifier()
118         clf = clf.fit(X_train_tfidf, training_polarity)
119         X_new_counts = count_vect.transform(test_texto)
120         X_new_tfidf = tfidf_transformer.transform(X_new_counts)
121         predicted = clf.predict(X_new_tfidf)
122         return predicted
```

Figura 40. Código para la clasificación de tweets mediante decision tree

En la *Figura 41* se presenta el código para el entrenamiento del clasificador, mediante el algoritmo *Random Forest* y posteriormente su predicción.

```
124     if op==4:  # RandomForest
125         from sklearn.ensemble import RandomForestClassifier
126         clf = RandomForestClassifier()
127         clf = clf.fit(X_train_tfidf, training_polarity)
128         X_new_counts = count_vect.transform(test_texto)
129         X_new_tfidf = tfidf_transformer.transform(X_new_counts)
130         predicted = clf.predict(X_new_tfidf)
131 
132         return predicted
```

Figura 41. Código para la clasificación de tweets mediante random forest

En la *Figura 42* se presenta el código para el entrenamiento del clasificador, mediante el algoritmo *K Neighbors Nearest* y posteriormente su predicción.

```
134     if op==5: # K Neighbors Nearest
135         from sklearn.neighbors.nearest_centroid import NearestCentroid
136         clf = NearestCentroid()
137         clf = clf.fit(X_train_tfidf, training_polarity)
138         X_new_counts = count_vect.transform(test_texto)
139         X_new_tfidf = tfidf_transformer.transform(X_new_counts)
140         predicted = clf.predict(X_new_tfidf)
141 
142         return predicted
```

Figura 42. Código para la clasificación de tweets mediante neighbors nearest

En la *Figura 43* se presenta el código para el entrenamiento del clasificador, mediante el algoritmo *Neural Network* y posteriormente su predicción.

```
145         if op ==6:  # Neural Neutwork
146             from sklearn.neural_network import MLPClassifier
147             clf = MLPClassifier()
148             clf = clf.fit(X_train_tfidf, training_polarity)
149             X_new_counts = count_vect.transform(test_texto)
150             X_new_tfidf = tfidf_transformer.transform(X_new_counts)
151             predicted = clf.predict(X_new_tfidf)
152 
153             return predicted
```

Figura 43. Código para la clasificación de tweets mediante neural network

En la *Figura 44* se presenta el código para realizar la clasificación de tweets mediante el método estadístico de léxico. En este código se presenta también el uso del diccionario de palabras, el cual contiene 4276 palabras, de ahí 1555 son positivas y 2721 son negativas. Para la clasificación de tweets en positivos o negativos, se realiza la contabilización de palabras, si en el tweet existe un mayor número de palabras positivas será positivo el tweet, si en el tweet el número de palabras negativas es mayor el será negativo. La polaridad de cada palabra es tomada del diccionario de palabras.

```
141 def lexico(test_texto):
142     import pandas as pd
143     diccionario= pd.read_csv('/var/www/html/ASentimientos/Process/diccionarioP.csv',
144         sep = ',', encoding='utf- 8', error_bad_lines=False)
145     val1 = "p"
146     val2 = "n"
147     cont = 0
148     data =[]
149     for tweet in test_texto:
150         words = tweet.split(' ')
151         cont_pos=0
152         cont_neg=0
153         for word in words:
154             contDic=0
155             for i in diccionario.word:
156                 if word.lower() ==i.lower() and diccionario.at[contDic, 'polarity'] ==val1:
157                     cont_pos = cont_pos + 1
158                 if word.lower() ==i.lower() and diccionario.at[contDic, 'polarity'] == val2:
159                     cont_neg = cont_neg +1
160                 contDic=contDic+1
161         cont = cont +1
162         if cont_pos == cont_neg:
163             data.append(0)
164         if cont_pos > cont_neg:
165             data.append(0)
166         if cont_pos < cont_neg:
167             data.append(1)
168 
169     predicted = pd.DataFrame(data,columns=['polarity'])
170     return predicted.polarity
```

Figura 44. Código para la clasificación de tweets mediante léxico

En la *Figura 45* y *Figura 46* se muestra el código empleado para la clasificación de tweets, mediante la votación mayoritaria de todas las técnicas empleadas tales como: *Naïve Bayes, Random Forest, Decision Tree, Support Vector Machine, Neural Network, K Neighbors*

Nearest y Léxico. Para determinar la polaridad de cada tweet como: positivo o negativo, se consideró que 5 de 7 técnicas deben estar a favor de la polaridad, para asignar la polaridad a ese tweet.

```
def ensembler(naive,svm,decisionTree,ramdomForest,knearest,mlp,lexico,data_set,const):
    temp = pd.DataFrame(columns=('word','tweet', 'polaridad',
        'user', 'location','date','id_categoria','creation_date'))
    cont=0
    cont_aux=0
    data =[]
    for i in data_set.tweet:
        val_pos=0
        val_neg=0
        if naive[cont] ==0:
            val_pos = val_pos +1
        else:
            if naive[cont] ==1:
                val_neg = val_neg +1
        if svm[cont] == 0:
            val_pos = val_pos +1
        else:
            if svm[cont] == 1:
                val_neg = val_neg +1
        if decisionTree[cont] == 0:
            val_pos = val_pos +1
        else:
            if decisionTree[cont] == 1:
                val_neg = val_neg + 1
        if ramdomForest[cont] == 0:
            val_pos = val_pos +1
        else:
            if ramdomForest[cont] == 1:
                val_neg = val_neg +1
```

Figura 45. Código para la clasificación de tweets mediante ensemble. (Parte a)

```
    if val_pos >= const:
        temp.loc[len(temp)] = [data_set.at[cont_aux,'word'],data_set.at[cont_aux,'tweet'],
        0,data_set.at[cont_aux,'user'],data_set.at[cont_aux,'location'],data_set.at[cont_aux,'date'],
        data_set.at[cont_aux,'id_categoria'],data_set.at[cont_aux,'creation_date']]
        cont_aux = cont_aux +1
    else:
        if val_neg >= const:
            temp.loc[len(temp)] = [data_set.at[cont_aux,'word'],data_set.at[cont_aux,'tweet'],
            1,data_set.at[cont_aux,'user'],data_set.at[cont_aux,'location'],
            data_set.at[cont_aux,'date'],data_set.at[cont_aux,'id_categoria'],
            data_set.at[cont_aux,'creation_date']]
            cont_aux = cont_aux +1

    cont=cont+1

return temp
```

Figura 46. Código para la clasificación de tweets mediante ensemble. (Parte b)

4.4. Modelo de Datos

En esta sección se presenta los modelos de datos usados para el desarrollo del sistema de análisis de sentimientos en Twitter. Dentro de la capa de datos se ha seleccionado ha MongoDB y a MySQL como bases de datos, así como se muestra en la sección 3.3.2.

La base de datos MongoDB fue seleccionada para acceder y almacenar de forma rápida, grandes números de datos como, los tweets extraídos. A diferencia de la base de datos MySQL que fue seleccionada para el almacenamiento y acceso de poca información, como es la polaridad de un tema relevante. Cada uno de estos modelos de datos se comunican directamente con la capa de proceso, para posteriormente mostrar los resultados en la capa de presentación. A continuación, se presenta la estructura de cada una de las bases de datos y la funcionalidad que desempeñaron en el desarrollo del sistema.

- MongoDB

Para el presente trabajo se utilizó el sistema de Base de Datos NoSQL de MongoDB para almacenar los tweets extraídos mediante el API de Twitter, el modelo implementado se menciona en la Figura 47. La tabla representa una base de datos donde: en la primera fila se encuentra el nombre de la base de datos, en la segunda el nombre de la colección y en la tercera los campos con su tipo. La "U" indica que el campo es único y el "*" indica que el campo es requerido.

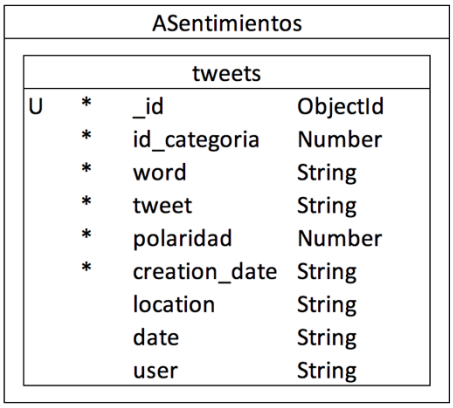

Figura 47. Modelo de Datos de MongoDB

- MySQL

Para la implementación, se consideró también el uso del sistema de Base de Datos relacional MySQL, para almacenar los datos preliminares de polaridad mostrados al usuario final, el modelo implementado se muestra en la *Figura 48*. Este modelo cuenta con 2 tablas: resumen y categoría, la tabla resumen cuenta con columnas: id_categoria, valor_positivo, valor_negativo y fecha_tweet, esta tabla contiene los datos preliminares de consulta; en cambio en la tabla categoría almacena 4 categorías predefinidas tales como: "Ecuador", "Moreno", "Galápagos" y "Correa". La relación de estas tablas es de una a muchos.

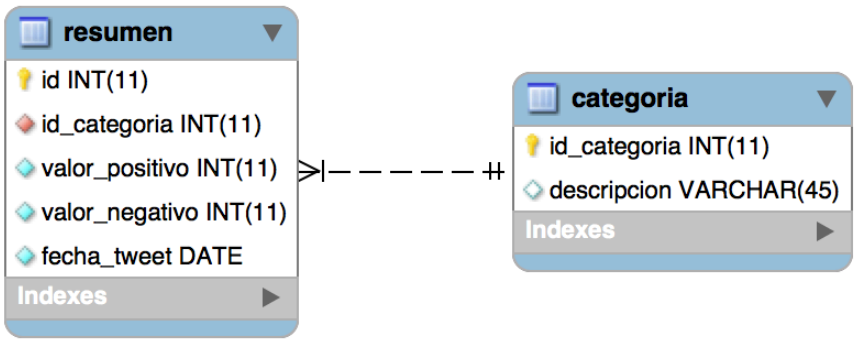

Figura 48. Modelo Entidad Relación de MySQL

4.5. Conclusiones

La implementación de herramientas de código abierto ha permitido que la solución en temas de costo no sea afectada, aunque dentro de las herramientas seleccionadas tenemos al lenguaje de programación Swift, el cual es propietario. El desarrollo del sistema mantiene un balance de componentes, así como una variedad de herramientas, haciendo al sistema multilenguaje y multiplataforma.

Los paquetes de Python usados para el análisis de sentimientos en la red social Twitter son fáciles de implementar, además de que cada uno de los paquetes relacionales al: procesamiento de lenguaje natural, extracción de tweets y *machine learning*, se encuentran bien documentados.

El uso de base de datos MongoDB ha permitido un acceso rápido, a los tweets almacenados en la base de datos y que sirven para el aprendizaje de cada uno de los clasificadores. La implementación de una base de datos relacional como es MySQL a diferencia de una

NoSQL, ha permitido que se realicen consulta SQL, de manera ágil hacia una tabla de poco volumen de datos.

En el siguiente capítulo se procederá con pruebas del sistema a nivel de: código, funcionalidad y clasificación de tweets, una vez que se ha realizado la implementación y detalle de cada una de las herramientas empleadas para el desarrollo de la propuesta.

5. Pruebas y Resultados

5.1. Introducción

En este capítulo se discute la experimentación con el sistema, esto mediante la realización de pruebas que establezcan parámetros y permitan evaluar el desempeño; para llevar a cabo este objetivo, se han empleado dos tipos de pruebas. En la primera prueba se realizará una comparación entre: la propuesta de este trabajo usando una técnica ensemble frente a servicios disponibles en internet, como es el caso de los APIs, además se efectuarán pruebas con diferentes *datasets* y técnicas de clasificación texto mencionadas en la sección 3.2.3, esto con el fin de determinar el comportamiento de cada técnica empleada. La segunda prueba está enfocada en el desarrollo del aplicativo móvil, en donde se tienen 2 pruebas: unitarias y de aceptación; para que finalmente se presente el análisis y discusión de las pruebas realizadas.

5.2. Pruebas correspondientes a la clasificación de tweets

Esta sección se presentan varios tipos de pruebas relacionadas al análisis de sentimientos; dentro de las pruebas tenemos: uso de diferentes *datasets*, uso de técnicas de preprocesamiento de texto y finalmente técnicas de clasificación de texto. En las siguientes secciones detallarán cada una de ellas.

5.2.1. Prueba de clasificación mediante el uso del dataset Moreno

Con la variedad de *datasets* que existen, así como los mencionados en la sección 2.3.1. Se propone realizar pruebas de desempeño para cada una de las técnicas empleadas en la sección 3.2.3, sobre determinado *datasets*, los cuales tengan las siguientes características: idioma

español, localidad ecuador y años de extracción 2017-2018. Con el fin de conocer el comportamiento de cada una de las técnicas, sobre el *dataset* empleado. Las pruebas se han dividido en dos: eliminar hashtags y sin eliminación de hashtags dentro del contenido de los tweets.

- **Objetivos de la prueba:**
    - Implementar técnicas de preprocesamiento de texto sobre los tweets.
    - Comparar los métodos de *machine learning* usados para la clasificación de tweets, mediante las métricas de evaluación.
    - Comparar los resultados obtenidos para el conjunto de datos de entrenamiento y de prueba.
- Descripción General de los datos:

Los tweets usados para la realización de las pruebas fueron extraídos en el año 2017 durante la campaña electoral hacia la Presidencia del Ecuador, específicamente los tweets se centran en el candidato Lenin Moreno, el cual es del partido político Alianza País. Adicionalmente se tiene un grupo pequeño de tweets que es del año 2018.

En las pruebas se han implementado dos tipos de *dataset*, con diferentes tamaños de datos, así como el tiempo en el que fueron extraídos los tweets. Los *dataset* se mencionan en las posteriores secciones de este documento.

- DataSet 1:

Para el primer caso se usó un total de 1718 tweets de los cuales 859 son positivos y 859 son negativos; dentro del total se ha establecido que el 70% de los datos sean para entrenamiento (1201 tweets de los cuales 601 son positivos y 600 son negativos) y el 30% restante sean para prueba (516 tweets, de los cuales 258 son positivos y 258 son negativos).

- DataSet 2:

El segundo caso de análisis se ha implementado un *dataset* que cuenta, con un conjunto de entrenamiento con 1718 tweets, de los cuales 859 son positivos y 859 son negativos, estos tweets fueron extraídos en el 2017 antes de las elecciones electorales para presidente de la República del Ecuador. También este *dataset* cuenta con datos de prueba, con un total de 100 tweets, de los cuales 50 son positivos y 50 son negativos, estos datos fueron extraídos en el presente año 2018.

- Procesamiento de Texto

Para este tipo de pruebas se han utilizado varias técnicas de procesamiento de texto sobre los tweets, esto para reducir el tamaño de texto y tener información más destacable durante el análisis; acontinuación, se muestras las técnicas usadas:

  - Remover WWW dentro del tweet.
  - Remover Stopwords en el tweet.
  - Remover htttp dentro del tweet.
  - Remover los hashtags.
  - Remover signos de puntuación.
  - Remover múltiples espacios en blanco.

- **Selección de Características**

Para la selección de características del tweet se usó $tf - idf$ el cual significa frecuencia de término por frecuencia de documento inversa. El objetivo de usar tf-idf en lugar de las frecuencias ocurrentes de un token en un documento, es reducir el impacto de tokens que ocurren con mucha frecuencia en un corpus dado y que, por lo tanto, son empíricamente menos informativas que las características que ocurren en un documento determinado.

- **Clasificación de Tweets**

Para las pruebas se utilizó una técnica meta, donde agrupa técnicas de *machine learning* (*Naïve Bayes, Random Forest, KNN, NN, SVM*) y de léxico respectivamente, esto para la clasificación de tweets en positivos y negativos. Adicional a estas técnicas tenemos algunas variantes; la primera variante es con el top 5 con los mejores métodos que tiene el mayor F1 y la segunda variante es el top 4 con los mejores que tienen el mayor F1.

Ensemble: En este método se determina la polaridad del tweet ya sea "Positivo" o "Negativo" mediante la votación mayoritaria de los métodos de *machine learning* y léxico.

  - Ensemble Todos

Para el cálculo de este tipo de Ensemble se usó los 7 clasificadores para determinar por mayoría si un tweet es positivo o negativo.

  - Ensemble Top 5 y Ensemble Top4

En el caso del Ensemble top 5 se seleccionó, los 5 clasificadores con más alto F1 y al Ensemble top 4 se selección, los 4 clasificadores con más alto F1.

A continuación, se muestra las*Tabla 10* con las comparativas de los *dataset* frente a cada método de clasificación usados, así como los resultados obtenidos mediante las métricas: *recall*, *precision* y F1.

DataSet 1			
Método	Recall	Precision	F1
Naïve Bayes	88.37%	78.62%	83.21%
Neural networks	81.40%	84.00%	82.68%
SVM	79.07%	85.36%	82.09%
KNN	68.60%	83.10%	75.16%
Random Forest	75.19%	70.80%	72.93%
Léxico	90.70%	56.80%	69.85%
Decision Tree	65.12%	71.49%	68.15%
Ensemble Todos	74.03%	87.21%	80.08%
Ensemble Top 5	80.62%	83.20%	81.89%
Ensemble Top 4	84.88%	82.64%	83.75%

DataSet 2			
Método	Recall	Precision	F1
Naïve Bayes	78.00%	78.00%	78.00%
Random Forest	78.00%	70.91%	74.29%
Neural networks	68.00%	77.27%	72.34%
Decision Tree	74.00%	69.81%	71.84%
Léxico	92.00%	56.10%	69.70%
SVM	60.00%	81.08%	68.97%
KNN	32.00%	72.73%	44.44%
Ensemble Todos	62.00%	77.50%	68.89%
Ensemble Top 5	86.00%	76.79%	81.13%
Ensemble Top 4	90.00%	69.23%	78.26%

Tabla 9. Resultados de la clasificación de los tweets para los dataset 1 y 2.

DataSet 1			
Método	Recall	Precision	F1
Naïve Bayes	88.37%	78.62%	83.21%
Neural networks	81.40%	83.67%	82.51%
SVM	79.07%	85.36%	82.09%
KNN	68.60%	83.10%	75.16%
Random Forest	68.99%	75.11%	71.92%
Léxico	90.70%	56.80%	69.85%
Decision Tree	61.63%	70.67%	65.84%
Ensemble Todos	72.48%	87.38%	79.24%
Ensemble Top 5	80.62%	84.21%	82.38%
Ensemble Top 4	84.88%	83.27%	84.07%

DataSet 2			
Método	Recall	Precision	F1
Naïve Bayes	78.00%	78.00%	78.00%
Random Forest	82.00%	68.33%	74.55%
Decision Tree	70.00%	72.92%	71.43%
Neural networks	66.00%	76.74%	70.97%
Léxico	92.00%	56.10%	69.70%
SVM	60.00%	81.08%	68.97%
KNN	32.00%	72.73%	44.44%
Ensemble Todos	66.00%	78.57%	71.74%
Ensemble Top 5	82.00%	73.21%	77.36%
Ensemble Top 4	84.00%	71.19%	77.06%

Tabla 10. Resultados de clasificación con dataset 1 y 2, sin eliminar hashtags

- **Discusión y análisis de resultados**

En la Tabla 9 y 10 se presentan los resultados obtenidos mediante la implementación de cada una de las técnicas de clasificación de tweets, entre las técnicas tenemos: *machine learning* y de léxico; adicional a estas técnicas tenemos el conjunto de las técnicas mediante una

votación mayoritaria “Ensemble”. Durante las pruebas se consideró la eliminación y no eliminación de hashtags dentro del contenido de cada tweet.

Los resultados obtenidos en la *Tabla 9* implementando la eliminación de hashtags para el *dataset* 1, nos indica que la técnica *naïve bayes* tiene el puntaje más alto en F1, frente a las demás técnicas con un valor de 83.21%, este valor es superado únicamente por el “Ensemble top 4” con 84.07%.

Los resultados obtenidos en la *Tabla 9* implementando la eliminación de hashtags para el *dataset* 2, nos indica que la técnica *naïve bayes* tiene el puntaje más alto en F1, frente a las demás técnicas con un valor de 78%.

Los resultados obtenidos en la *Tabla 10* sin implementar la eliminación de hashtags para el *dataset* 1, nos indica que la técnica *naïve bayes* tiene el puntaje más alto en F1, frente a las demás técnicas con un valor de 83.21%, este valor es superado únicamente por el “Ensemble top 4” con 84.07%.

Los resultados obtenidos en la *Tabla 10* sin implementar la eliminación de hashtags para el *dataset* 2, nos indica que la técnica *naïve bayes* tiene el puntaje más alto en F1, frente a las demás técnicas con un valor de 78%.

Los resultados de usar un *dataset* del 2017 y 2018 son notorios, llegando a concluir que el *dataset* que se usó afectará directamente en el desempeño de cada uno de los clasificadores, por esta razón es necesario considerar un *dataset* especializado en el análisis de sentimientos que se vaya a realizar, además de haber sido clasificado en un cierto tiempo, ya que los posts escritos por los usuarios de Twitter pueden variar con el tiempo.

Con respecto a la técnica de eliminar o no los hashtags dentro del contenido de cada uno de los tweets podemos mencionar que no afecta en gran media sobre el rendimiento de cada una de las técnicas *machine learning* y del “Ensemble”.

Las métricas de *recall* y *precision* obtenidas en las pruebas, son de un 70% a 80% respectivamente, estos valores nos indican que el proceso de predicción, para cada uno de los clasificadores tienen un 20% a 30% de error considerado como falsos positivos o falsos negativos. Estos valores de error no son tan altos considerando el uso de un *dataset* clasificado manualmente y de hace un año atrás.

5.2.2. Prueba de clasificación mediante APIs

Actualmente en el internet se dispone de varios servicios o herramientas, las cuales permiten realizar la clasificación de texto. Es por esta razón que se pretende realizar pruebas, con el fin de comparar el desempeño de estas herramientas frente a las técnicas empleadas en el trabajo. Entre las herramientas o APIs que se ha usado en las pruebas tenemos: Meaning Cloud[16] e IBM[17], estos APIs fueron seleccionados, por motivo de tener una versión libre, además de que son fáciles de implementar y son enfocados en el análisis de sentimientos.

- **Objetivos de la prueba:**
    - Implementar técnicas de preprocesamiento de texto sobre los tweets.
    - Comparar los métodos de *machine learning* con los APIs usados en el proceso de clasificación de tweets, mediante las métricas de evaluación.
- **Descripción General de los datos:**

Los tweets usados en esta prueba corresponden al *dataset* de "Moreno". El *dataset* contiene 1201 tweets de los cuales 501 son positivos y 506 son negativos.

- **Procesamiento de Texto**

Para este tipo de pruebas se han utilizado varias técnicas de procesamiento de texto sobre los tweets, esto para reducir el tamaño de texto y tener información más destacable durante el análisis; a continuación, se muestras las técnicas usadas:

- Remover WWW dentro del tweet.
- Remover *Stopwords* en el tweet.
- Remover htttp dentro del tweet.
- Remover los hashtags.
- Remover signos de puntuación.
- Remover múltiples espacios en blanco.

[16] https://www.meaningcloud.com/es/

[17] https://www.ibm.com/watson/services/natural-language-understanding/

- **Clasificación de Tweets**

Para la clasificación de los tweets se usó 6 métodos de *machine learning* tales como: *naïve bayes, random forest, knn, nn, svm*, 1 método léxico y 2 APIs. En la siguiente tabla podemos observar las métricas obtenidas para cada una de las técnicas usadas.

Método	Recall	Precision	F1
Naïve Bayes	88.37%	78.62%	83.21%
NN	81.40%	84.00%	82.68%
SVM	79.07%	85.36%	82.09%
KNN	68.60%	83.10%	75.16%
Random Forest	75.19%	70.80%	72.93%
Meaning Cloud	84.88%	62.93%	72.28%
Léxico	90.70%	56.80%	69.85%
IBM	94.90%	54.63%	69.34%
Decison Tree	65.12%	71.49%	68.15%

Tabla 11. Resultados de pruebas usando APIs y técnicas de machine learning

- **Discusión y análisis de resultados**

En la *Tabla 11* se presentan los resultados obtenidos, al usar técnicas de *machine learnig* y APIs para la clasificación de tweets; dentro de la tabla tenemos varias métricas tales como: *recall*, *precision* y F1. En el contenido de la tabla podemos observar que el método con mayor F1 es naïve bayes con 83.21% y la menor es *decision tree* con 68.15%. Esto cambia si analizamos la métrica *precision* donde la técnica SVM es la mayor con 85.36%, en términos de *recall* tenemos a NB como el valor más con 88.37%.

Los resultados de rendimiento entre los APIs y las técnicas de *machine leaning* son considerables; las técnicas de *machine learnig* son superiores a los APIS, considerando el uso gratuito de los APIs, así como los modelos genéricos que disponía cada uno de los APIs.

5.2.3. Prueba de clasificación mediante el uso del dataset Sentiment140

Con la variedad de *datasets* que existen, así como los mencionados en la sección 2.3.1. Se propone realizar pruebas de desempeño para cada una de las técnicas empleadas en la sección

2.3.4, sobre el *dataset* Sentiment140, ya que cuenta con las siguientes características: idioma inglés, gran cantidad de texto clasificado. Cabe mencionar que este *dataset* no fue implementado en el proyecto, debido a sus características.

Para las pruebas únicamente se usó 10000 comentarios del total, ya que los recursos de hardware fueron limitados. El *dataset* usado cuenta con 5000 comentarios positivos y 5000 negativos [18]. Las pruebas fueron efectuadas en la herramienta Weka[19].

- **Objetivos de la prueba:**
    - Implementar técnicas de preprocesamiento de texto sobre los comentarios.
    - Comparar los métodos de *machine learning* usados para la clasificación de tweets, mediante las métricas de evaluación.
    - Comparar los resultados obtenidos para el conjunto de datos de entrenamiento y de prueba.
- **Procesamiento de Texto**

Para este tipo de pruebas se han utilizado varias técnicas de procesamiento de texto sobre los comentarios, esto para reducir el tamaño de texto y tener información más destacable durante el análisis; a continuación, se muestras las técnicas usadas:

- *Term frecuency*
- *Stemmer*
- *Remover stopwords*
- *Lowercase*

- **Clasificación de comentarios**

Para la clasificación de los comentarios en: positivos y negativos, se usó 5 métodos de *machine learning* tales como: *naïve bayes, svm, knn, decision tree y random forest*.

[18] http://help.sentiment140.com/for-students/

[19] https://www.cs.waikato.ac.nz/ml/weka/

- **Evaluación de comentarios**

Para la evaluación de cada uno de los modelos se usó "Cross Validation k = 10", lo que significa que se realizó 10 iteraciones, para generar el dataset de entrenamiento y prueba; a continuación, se presentan los resultados obtenidos para cada uno de los clasificadores:

- Naïve Bayes

En la Tabla 12 se presentan 6 pruebas utilizando el método *naïve bayes* para la clasificación de tweets, estas pruebas contemplaron una configuración del clasificador por defecto y con kernel.

Descripción del parámetro:

Kernel Estimator: usa un estimador kernel para atributos numéricos en lugar de una distribución normal.

No	Descripción	Precision	Recall	F1	Atributos
1	NB c/ kernel	76.30%	69.16%	72.56%	2159
2	NB c/ kernel	76.66%	68.86%	72.55%	4158
3	NB c/ kernel	75.39%	68.36%	71.70%	1039
4	NB default	69.20%	62.20%	65.52%	4158
5	NB default	69.14%	62.16%	65.47%	2159
6	NB default	68.97%	61.76%	65.17%	1039

Tabla 12. Resultado de clasificación de comentarios usando naïve bayes

- Support Vector Machine

En la Tabla 13 se presentan 2 pruebas utilizando el método *support vector machine*, con una configuración por defecto y con "build calibrationmodels = true".

Descripción del parámetro:

BuildCalibrationModels: Si se ajusta la calibración del modelo para los resultados del SVM.

No	Descripción	Precision	Recall	F1	Atributos
2	SMO / Build calibrationmodels = true	74.25%	70.40%	72.27%	1039
1	SMO default	75.03%	68.74%	71.75%	1039

Tabla 13. Resultado de clasificación de comentarios usando svm

- KNN

En la *Tabla 14* se presentan 2 pruebas utilizando el método k vecinos más cercanos, con una configuración por defecto y con "k = 4 vecinos".

Descripción del parámetro:

K vecinos: La cantidad de vecinos a usar.

No	Descripción	Precision	Recall	F1	Atributos
1	IBK default	67.91%	59.52%	63.44%	1039
2	IBK / K = 4 Vecinos	70.63%	50.64%	58.99%	1039

Tabla 14. Resultado de clasificación de comentarios usando knn

- Decision Tree

En la *Tabla 15* se presentan 2 pruebas utilizando el método *decision tree*, con una configuración por defecto y con "lamplace = true, reduceerrorpruning = true".

Descripción de los parámetros:

Lamplace: si los conteos en las hojas se suavizan según Laplace.

ReduceError: la poda reduced-error está usando en la poda tipo C.4.5.

No	Descripción	Precision	Recall	F1	Atributos
1	Tree J48 default	70.20%	65.86%	67.96%	1039
2	Tree J48 / Lamplace = true / ReduceErrorPruning = true	69.61%	63.62%	66.48%	1039

Tabla 15. Resultado de clasificación de comentarios usando decision tree

- Random Forest

En la *Tabla 16* se presentan 2 pruebas utilizando el método *random forest*, con una configuración por defecto y con "seed = 10".

Descripción del parámetro:

Seed: El número aleatorio de semillas que se utilizará.

No	Descripción	Precision	Recall	F1	Atributos
2	Tree Random Forest / Seed = 10	73.50%	72.44%	72.97%	1039
1	Tree Random Forest / default	73.50%	72.16%	72.82%	1039

Tabla 16. Resultado de clasificación de comentarios usando random forest

- **Discusión y análisis de resultados**

Desde la *Tabla 12* a la *Tabla 16* se presentan los resultados obtenidos con cada una de las técnicas de *machine learning* usando el *dataset* Sentiment140; en estas tablas se puede observar la configuración realizada en el clasificador, así como las métricas obtenidas tales como *precision*, *recall* y F1; adicionalmente tenemos el número de atributos usados en cada prueba.

Los resultados obtenidos durante las pruebas indican que la técnica *tree random forest* tiene el mejor rendimiento, esto con un valor de F1 del 72.97%. En cambio, la técnica con el valor más alto de *precision* es *naïve bayes* con un 76.66% y finalmente la técnica con el valor más alto de *recall* es *tree random forest* con un 72.44%.

Algunas técnicas no presentan un número mayor de atributos a diferencia de otras, esto se debió a que muchas de las técnicas de *machine learning* tardan mucho tiempo en ser procesadas. Por este motivo se realizó solamente con la técnica *naïve bayes*, cambiar el número de atributos.

5.3. Pruebas del aplicativo móvil

Después de culminar con el desarrollo de un software, se debe verificar que se cumplan con las especificaciones planteadas desde un inicio por el analista o el propio cliente, y/o eliminar los posibles errores que se hayan cometido en cualquier etapa del desarrollo, para conseguir este objetivo se realizan pruebas como por ejemplo: unitarias, las cuales verifican el correcto funcionamiento de un módulo de código o también las pruebas de aceptación, las cuales determinar si un sistema satisface los criterios del usuario o cliente, para que finalmente se acepte o no el aplicativo. Por este motivo en esta sección se detallarán las pruebas unitarias y pruebas de aceptación.

5.3.1. Pruebas Unitarias

En este apartado se presentará las pruebas unitarias relacionadas, con los servicios web disponibles para el cliente o capa de presentación; las pruebas realizadas contemplan la funcionalidad de cada uno de los servicios que tiene el sistema de análisis de sentimientos. La herramienta empleada para pruebas es Postman, la cual permitió realizar la petición a cada servicio web.

- **Objetivos de la prueba:**
  - Implementar técnicas de preprocesamiento de texto sobre los tweets.
  - Comparar los métodos de *machine learning* usados para la clasificación de tweets, mediante las métricas de evaluación.
  - Comparar los resultados obtenidos para el conjunto de datos de entrenamiento y de prueba.
- **Pruebas del web service consulta online:**
  - Primera petición al web service

URL	http://uceipl.ddnsking.com/ASentimientos/public/instantaneo/Cuenca
Parámetro	Cuenca
Código de Respuesta	200
Respuesta JSON	{"results":[{"tema":"Cuenca","total":35,"positivo":6,"negativo":29}]}
Tiempo de Respuesta (minutos)	2,47

Tabla 17. Web Service - consulta online primera petición

  - Segunda petición al web service

URL	http://uceipl.ddnsking.com/ASentimientos/public/instantaneo/Cuenca
Parámetro	Cuenca
Código de Respuesta	200
Respuesta JSON	{"results":[{"tema":"Cuenca","total":35,"positivo":6,"negativo":29}]}
Tiempo de Respuesta (minutos)	0.21

Tabla 18. Web Service - consulta online segunda petición

- **Pruebas del web Service consulta predefinida:**

URL	http://uceipl.ddnsking.com/ASentimientos/public/ecuador/25
Parámetro	25
Código de Respuesta	200
Respuesta JSON	{"results":[{"positivo":50,"negativo":0,"fecha":"2018-05-17"},{"positivo":50,"negativo":0,"fecha":"2018-05-18"},{"positivo":50,"negativo":0,"fecha":"2018-05-19"},{"positivo":50,"negativo":0,"fecha":"2018-05-20"},{"positivo":50,"negativo":0,"fecha":"2018-05-21"},{"positivo":50,"negativo":0,"fecha":"2018-05-22"},{"positivo":50,"negativo":0,"fecha":"2018-05-23"},{"positivo":50,"negativo":0,"fecha":"2018-05-24"},{"positivo":50,"negativo":0,"fecha":"2018-05-25"},{"positivo":50,"negativo":0,"fecha":"2018-05-26"},{"positivo":50,"negativo":0,"fecha":"2018-05-27"}]}
Tiempo de Respuesta (minutos)	0,01

Tabla 19. Web Service - consulta offline predefinida - tema: "Ecuador"

URL	http://uceipl.ddnsking.com/ASentimientos/public/moreno/25
Parámetro	25
Código de Respuesta	200
Respuesta JSON	{"results":[{"positivo":0,"negativo":45,"fecha":"2018-05-17"},{"positivo":1,"negativo":43,"fecha":"2018-05-18"},{"positivo":0,"negativo":46,"fecha":"2018-05-19"},{"positivo":5,"negativo":38,"fecha":"2018-05-20"},{"positivo":0,"negativo":42,"fecha":"2018-05-21"},{"positivo":0,"negativo":45,"fecha":"2018-05-22"},{"positivo":2,"negativo":44,"fecha":"2018-05-23"},{"positivo":2,"negativo":45,"fecha":"2018-05-24"},{"positivo":1,"negativo":43,"fecha":"2018-05-25"},{"positivo":1,"negativo":45,"fecha":"2018-05-26"},{"positivo":1,"negativo":47,"fecha":"2018-05-27"}]}
Tiempo de Respuesta (minutos)	0,01

Tabla 20. Web Service - consulta offline predefinida – tema "Moreno"

URL	http://uceipl.ddnsking.com/ASentimientos/public/galapagos/25
Parámetro	25
Código de Respuesta	200
Respuesta JSON	{"results":[{"positivo":1,"negativo":46,"fecha":"2018-05-17"},{"positivo":4,"negativo":45,"fecha":"2018-05-18"},{"positivo":2,"negativo":48,"fecha":"2018-05-19"},{"positivo":2,"negativo":46,"fecha":"2018-05-20"},{"positivo":2,"negativo":47,"fecha":"2018-05-21"},{"positivo":1,"negativo":49,"fecha":"2018-05-22"},{"positivo":1,"negativo":49,"fecha":"2018-05-23"},{"positivo":0,"negativo":50,"fecha":"2018-05-24"},{"positivo":0,"negativo":50,"fecha":"2018-05-25"},{"positivo":1,"negativo":49,"fecha":"2018-05-26"},{"positivo":0,"negativo":50,"fecha":"2018-05-27"}]}
Tiempo de Respuesta (minutos)	0,01

Tabla 21. Web Service - consulta offline predefinida - tema: “Galápagos”

URL	http://uceipl.ddnsking.com/ASentimientos/public/correa/25
Parámetro	25
Código de Respuesta	200
Respuesta JSON	{"results":[{"positivo":1,"negativo":49,"fecha":"2018-05-17"},{"positivo":0,"negativo":50,"fecha":"2018-05-18"},{"positivo":0,"negativo":50,"fecha":"2018-05-19"},{"positivo":0,"negativo":50,"fecha":"2018-05-20"},{"positivo":0,"negativo":50,"fecha":"2018-05-21"},{"positivo":0,"negativo":50,"fecha":"2018-05-22"},{"positivo":0,"negativo":50,"fecha":"2018-05-23"},{"positivo":0,"negativo":50,"fecha":"2018-05-24"},{"positivo":0,"negativo":50,"fecha":"2018-05-25"},{"positivo":0,"negativo":50,"fecha":"2018-05-26"},{"positivo":0,"negativo":50,"fecha":"2018-05-27"}]}
Tiempo de Respuesta (minutos)	0,01

Tabla 22. Web Service - consulta offline predefinida - tema: “Correa”

- **Discusión y análisis de resultados**

Los resultados presentados en las Tabla 17 y Tabla 18 indican que la funcionalidad sobre la consulta online se encuentra disponible y sin problemas; también podemos observar que el tiempo de espera durante la primera petición es de 2,47 minutos es superior a la segunda consulta que es de 0,21 minutos; esta diferencia se debe al contenido del *dataset*, ya que, por primera vez no existen tweets dentro del *dataset* relacionados a la palabra de consulta, eso no ocurre cuando se lo realiza por segunda vez, donde el proceso de análisis de sentimientos, ya cuenta con un *dataset* relacionado a la palabra de búsqueda.

Los resultados presentados en las Tabla 19, Tabla 20, Tabla 21 y Tabla 22, indican que la funcionalidad sobre la consulta online, se encuentra disponible y sin problemas. En cuanto al tiempo de respuesta al usuario final, cada uno de los servicios web se demoran 0,01 minuto, considerado un tiempo aceptable.

5.3.2. Pruebas de Aceptación

En esta sección se realizarán pruebas de aceptación con el cliente, donde se probará el software y verificará que cumpla con las expectativas del usuario. Estas pruebas generalmente son funcionales y se basan en los requisitos definidos por el cliente.

Las pruebas se realizaron con 5 personas, a cada persona se le formulo las siguientes preguntas:

- Usuario # 1

No	Preguntas	Respuestas
1	¿Qué puede decir de Ecuador?	Presenta valores altos de aceptacion casi un 90%
2	¿Qué puede decir de Moreno?	Presenta valores bajos de aceptacion 90%
3	¿Qué puede decir de las Islas Galápagos?	Presenta valores bajos de aceptacion 90%
4	¿Qué puede decir de Correa?	Presenta valores bajos de aceptacion 100%
5	¿Qué puede decir de Portugal?	Presenta una polaridad positiva
6	¿Es entendible para el usuario?	Si
7	¿La información presentada te ayudo?	Si

Tabla 23. Cuestionario para el usuario #1

- Usuario # 2

No	Preguntas	Respuestas
1	¿Qué puede decir de Ecuador?	Tiene una polaridad postiva
2	¿Qué puede decir de Moreno?	Tiene una polaridad negativa
3	¿Qué puede decir de las Islas Galápagos?	Tiene una polaridad postiva
4	¿Qué puede decir de Correa?	Tiene una polaridad negativa
5	¿Qué puede decir de Portugal?	Que tiene una polaridad positiva
6	¿Es entendible para el usuario?	Si
7	¿La información presentada te ayudo?	No, esto debido a que algunos temas ya conocia con anterioridad su polaridad, exepto portugal.

Tabla 24. Cuestionario para el usuario #2

- Usuario # 3

No	Preguntas	Respuestas
1	¿Qué puede decir de Ecuador?	Que se habla muy bien
2	¿Qué puede decir de Moreno?	Que hablan muy mal de esta persona
3	¿Qué puede decir de las Islas Galápagos?	Que las personas hablan muy bien
4	¿Qué puede decir de Correa?	Que hablan muy mal de esta persona
5	¿Qué puede decir de Portugal?	Que hablan muy bien de Portugal
6	¿Es entendible para el usuario?	Si
7	¿La información presentada te ayudo?	Si, ya que no tenia conocimiento de los temas.

Tabla 25. Cuestionario para el usuario #3

- Usuario # 4

No	Preguntas	Respuestas
1	¿Qué puede decir de Ecuador?	Presenta valores altos de aceptacion casi un 90%
2	¿Qué puede decir de Moreno?	Presenta valores bajos de aceptacion 90%
3	¿Qué puede decir de las Islas Galápagos?	Presenta valores bajos de aceptacion 90%
4	¿Qué puede decir de Correa?	Presenta valores bajos de aceptacion 100%
5	¿Qué puede decir de Portugal?	Presenta una polaridad positiva
6	¿Es entendible para el usuario?	Si
7	¿La información presentada te ayudo?	Si

Tabla 26. Cuestionario para el usuario #4

- Usuario # 5

No	Preguntas	Respuestas
1	¿Qué puede decir de Ecuador?	Presenta valores altos de aceptacion casi un 90%
2	¿Qué puede decir de Moreno?	Presenta valores bajos de aceptacion 90%
3	¿Qué puede decir de las Islas Galápagos?	Presenta valores bajos de aceptacion 90%
4	¿Qué puede decir de Correa?	Presenta valores bajos de aceptacion 100%
5	¿Qué puede decir de Portugal?	Presenta una polaridad positiva
6	¿Es entendible para el usuario?	Si
7	¿La información presentada te ayudo?	Si

Tabla 27. Cuestionario para el usuario #5

- **Discusión y análisis de resultados**

Las preguntas que se le hacen a los usuarios son de forma general lo que le da una perspectiva global del uso y funcionamiento del aplicativo desarrollado en esta investigación.

Todos los usuarios respondieron "SÍ" a la pregunta 6, indicando que el aplicativo móvil es fácil de comprender.

Los usuarios presentaron inconvenientes en la gráfica de barras, ya que el intervalo de tiempo no es claro, aun así, cada usuario supo interpretar el grafico y respondió a las preguntas 1 y 4. Esto se puede verificar desde la tabla *Tabla 23* a la *Tabla 27*.

Los usuarios no presentaron inconvenientes con la gráfica de pastel y pudieron rápidamente interpretar los valores mostrados. Esto se puede verificar en las respuestas a la pregunta 5 de la *Tabla 23* a la *Tabla 27*.

5.4. Conclusiones

La técnica de eliminación y no eliminación de hashtags implementada en la sección 5.2.1, no afecta en gran medida al desempeño de cada técnica de clasificación, esto puede ser causado por el tamaño del tweet, el cual es pequeño. Considerando esto podrían variar los resultados con textos más grandes.

Dentro de las pruebas realizadas por los APIs, se puede mencionar que no superan a las técnicas de *machine learnig*, considerando que estos servicios son gratuitos y genéricos a diferencias de los clasificadores, los cuales aprendieron y predijeron los tweets sobre el *dataset* de “Moreno”.

Los tiempos de respuesta para la consulta online, son muy largos a lo establecido por el usuario, esto debido a ciertas características de la arquitectura como es: hardware limitado, conexión de internet y la programación estructurada.

6. Conclusiones y trabajo futuro

El análisis de sentimientos permite a través del procesamiento de lenguaje natural tener una visión aproximada de lo que las personas están opinando en un momento dado.

Los resultados obtenidos del análisis de sentimientos realizado con la herramienta no son exactos, puesto que al analizar el texto no se puede determinar si el usuario que escribe lo está haciendo de una forma sarcástica o no, sin embargo, dan una visión aproximada de lo que los usuarios de la plataforma Twitter están hablando sobre un tema en particular.

El tiempo de respuesta del análisis lo ocupa en gran parte la descarga de Tweets, puesto que depende del API y las restricciones de Twitter para realizar este proceso, debido a esto se debe limitar el número de Tweets a descargar para obtener un tiempo razonable de espera al usuario final.

Cuando se varían las claves del API de Twitter para cada análisis se evita que se alcance el número máximo de Tweets a descargar permitido por Twitter, de esta forma la herramienta tiene la capacidad de realizar análisis de varios usuarios de manera concurrente.

Como ya se ha mencionado, el rendimiento obtenido ha sido de un 80 % en el sistema de análisis de sentimientos propuesto; aunque, este resultado se puede aceptar de manera optimista, pues se trata de un trabajo de puesta a punto de las técnicas de AS. Las técnicas de *machine learning* y de léxico, usadas en el *dataset* "Moreno" tanto en la eliminación como no eliminación de hashtags. Presentan valores altos en rendimiento, ya sea de forma individual o en conjunto.

Primeramente, uno de los motivos principales para tener un rendimiento alto, es la utilización de un *dataset* clasificado manualmente y que cuenta con varias características destacables como son: el periodo de los tweets, la localización de los tweets y la temática que se abordó. Esto ha permitido tener un *dataset* robusto y aproximado a la realidad.

Un segundo motivo para tener estos valores es el uso de un *dataset* clasificado a mano, por lo que tiene un alto grado de fiabilidad de cada uno de los tweets que fueron clasificados.

Un tercer motivo es la combinación de técnicas, permitiendo así mejorar las debilidades de ciertas técnicas frente a otras, y de esta manera tener mejor performance.

Otro de los motivos causantes de esta exactitud es posiblemente la tendencia a expresarnos usando palabras con orientación positiva, precedidas por una negación para expresar sentimientos negativos, por lo que los clasificadores se confunden al usar BOW. Sin duda con la implementación de bigramas en las líneas futuras se resolverá este problema y quizás aumente la exactitud.

La técnica léxico también destaco con un alto F1, a pesar tener un diccionario de palabras muy genérico y no especializado. En temas de diccionario también debemos mencionar, que no se consideraron modismo de cada localidad, ya que son propias de cada país o región.

6.1. Principales aportaciones

Este estudio supone una primera aproximación al AS en español; lamentablemente la mayoría *datasets* y estudios se encuentran en inglés lo que supone una gran desventaja. Gracias al corpus "Moreno" se ha podido obtener algunos clasificadores bastantes robustos, pero con una capacidad de mejora.

- Se ha demostrado mediante las pruebas que la eliminación o no de los hashtags, no mejoran significativamente el rendimiento de los clasificadores.
- Se ha implementado una Meta técnica, la cual combina algoritmos de *machine learning* y de lexico, para el análisis de sentimientos; esta combinación ha permitido un mejor rendimiento en clasificación de texto.
- Tras finalizar la búsqueda de tweets y analizar las polaridades, se despliega un gráfico de barras, el cual muestra el porcentaje de tweets negativos y positivos durante la ejecución.

6.2. Líneas futuras

Mediante el análisis de resultados, las conclusiones y teniendo en cuenta el crecimiento en el desarrollo del procesamiento del lenguaje natural, en parte gracias al avance de las nuevas tecnologías como el Big Data; a continuación, se van a describir las futuras líneas de investigación.

Las ampliaciones futuras de este estudio se enfocarán en tareas a corto plazo y largo plazo. Poco a poco se irá mejorando el sistema hasta subir la *precision*, el *recall* y F1.

Dentro de las tareas **a corto plazo:**

- Añadir la funcionalidad de separar en Positivo/Negativo/Neutral para aquellos tweets con una polaridad muy poco intensa.
- Utilizar bigramas y trigramas en vez de solo unigramas.
- Mejorar los equipos usados para reducir los tiempos de ejecución y no saturar el procesador del ordenador, pues no es el más adecuado para trabajar con archivos pesados.
- Implementar una ejecución en paralelo durante la predicción de cada uno de modelos usados, esto para reducir de los tiempos de procesamiento; se podría usar Apache Spark.
- Eliminar las palabras neutras o sin sentido de los vectores de características para quedarnos con aquellas que si poseen sentimiento; eliminar los términos que contienen poca información aporta claridad, reduce el ruido y disminuye la dimensionalidad por lo que inevitablemente la exactitud mejorará.
- Implementar otro tipo de métodos para la selección de características tales como: TF, *Chi square*, entre otros.

En cuanto a las tareas a **largo plazo.**

- Desarrollar una plataforma web/servidor donde almacenar en tiempo real los tweets analizados clasificados acorde a su polaridad y visualizar las estadísticas más descriptivas.
- Incrementar el tamaño del *dataset* "Moreno" esto con la posibilidad de efectuar un entrenamiento de cada clasificador, usando un número mayor de tweets que el que está usando actualmente.

Referencias

[1] A. E. Ayush Kumar, Vamsi Krishna Akella, *IITPSemEval: Sentiment Discovery from 140 Characters*, SemEval. 2015.

[2] R. W. Ruth Talbot, Chloe Acheampong, *SWASH: A Naive Bayes Classifier for Tweet Sentiment Identification*. 2015.

[3] M. L. Zhihua Zhang, GuoshunWu, *ECNU: Multi-level Sentiment Analysis on Twitter Using Traditional Linguistic Features andWord Embedding Features*. 2015.

[4] S. H. M. Savitha Mathapati and V. K. R, "Sentiment Analysis and Opinion Mining from Social Media : A Review," vol. 16, no. 5, 2016.

[5] S. Rill, D. Reinel, J. Scheidt, and R. V. Zicari, "PoliTwi: Early detection of emerging political topics on twitter and the impact on concept-level sentiment analysis," *Knowledge-Based Syst.*, vol. 69, no. 1, pp. 24–33, 2014.

[6] L. Dang-Xuan and S. Stieglitz, "Impact and Diffusion of Sentiment in Political Communication–An Empirical Analysis of Political Weblogs," *AAAI Conf. Weblogs Soc. Media*, pp. 3500–3509, 2012.

[7] Y. Yu, W. Duan, and Q. Cao, "The impact of social and conventional media on firm equity value: A sentiment analysis approach," *Decis. Support Syst.*, vol. 55, no. 4, pp. 919–926, 2013.

[8] X. Li, H. Xie, L. Chen, J. Wang, and X. Deng, "News impact on stock price return via sentiment analysis," *Knowledge-Based Syst.*, vol. 69, no. 1, pp. 14–23, 2014.

[9] J. Smailović, M. Grčar, N. Lavrač, and M. Žnidaršič, "Stream-based active learning for sentiment analysis in the financial domain," *Inf. Sci. (Ny).*, vol. 285, no. 1, pp. 181–203, 2014.

[10] N. K. Cobb, D. Mays, and A. L. Graham, "Sentiment analysis to determine the impact of online messages on smokers' choices to use varenicline," *J. Natl. Cancer Inst. - Monogr.*, no. 47, pp. 224–230, 2013.

[11] T. H. Nguyen, K. Shirai, and J. Velcin, "Sentiment analysis on social media for stock movement prediction," *Expert Syst. Appl.*, vol. 42, no. 24, pp. 9603–9611, 2015.

[12] R. Duwairi and M. El-Orfali, “A study of the effects of preprocessing strategies on sentiment analysis for Arabic text,” *J. Inf. Sci.*, vol. 40, no. 4, pp. 501–513, 2014.

[13] R. J. McGee and R. L. Warms, “Theory in Social and Cultural Anthropology. An Encyclopedia,” p. 996, 2013.

[14] M. Moreno, “Cómo triunfar en las Redes Sociales,” 2015.

[15] C. Zhang, J. Sun, X. Zhu, and Y. Fang, “Privacy and Security for Online Social Networks: Challenges and Opportunities,” *IEEE Netw.*, vol. 24, no. 4, pp. 13–18, 2010.

[16] N. B. E. danah m. boyd, “Social Network Sites: Definition, History, and Scholarship,” *Int. Rev. Res. Open Distance Learn.*, vol. 12, no. 3, pp. 210–230, 2007.

[17] S. P. Sivasubramanian and N. Suganya, “Sentiment Analysis On Micro-blogs,” pp. 46–51, 2017.

[18] K. Meek, M. Barnes, and C. Hanson, “The Impact of Online Social Capital on Twitter Users At-risk for Suicide,” p. 5090, 2017.

[19] D. Garcia, P. Mavrodiev, D. Casati, and F. Schweitzer, “Understanding Popularity, Reputation, and Social Influence in the Twitter Society,” *Policy and Internet*, vol. 9, no. 3, pp. 343–364, 2017.

[20] S. Alowaidi, M. Saleh, and O. Abulnaja, “Semantic Sentiment Analysis of Arabic Texts,” *IJACSA) Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 2, pp. 256–262, 2017.

[21] M. VOHRA and J. TERAIYA, “A Comparative Study of Sentiment Analysis Techniques,” *Ejournal.Aessangli.in*, vol. 17, no. 4, pp. 313–317, 2013.

[22] G. Angiani *et al.*, “A comparison between preprocessing techniques for sentiment analysis in Twitter,” *CEUR Workshop Proc.*, vol. 1748, no. Ml, 2016.

[23] B. Liu, *Sentiment Analysis and Subjectivity*. 2010.

[24] W. Medhat, A. Hassan, and H. Korashy, “Sentiment analysis algorithms and applications: A survey,” *Ain Shams Eng. J.*, vol. 5, no. 4, pp. 1093–1113, 2014.

[25] V. A. Kharde and S. S. Sonawane, “Sentiment Analysis of Twitter Data: A Survey of Techniques,” *Int. J. Comput. Appl.*, vol. 139, no. 11, pp. 975–8887, 2016.

[26] A. K. Behera, “Performance Analysis of Supervised Machine Learning Techniques

for Sentiment Analysis," pp. 128–133, 2017.

[27] H. M. Zin, N. Mustapha, M. A. A. Murad, and N. M. Sharef, "The effects of pre-processing strategies in sentiment analysis of online movie reviews," *AIP Conf. Proc.*, vol. 1891, 2017.

[28] M. Z. Asghar, A. Khan, S. Ahmad, and F. M. Kundi, "A Review of Feature Extraction in Sentiment Analysis," *J. Basic. Appl. Sci. Res*, vol. 4, no. 3, pp. 181–186, 2014.

[29] C. Lin and Y. He, "Joint sentiment/topic model for sentiment analysis," in *Proceeding of the 18th ACM conference on Information and knowledge management - CIKM '09*, 2009, p. 375.

[30] R. M. Eugenio Martinez-Camara, M.Teresa Martin-Valdivia, L.Alfonso Ureña-Lopez, "Polarity classification for Spanish tweets using the COST corpus," *J. Intell. Mater. Syst. Struct.*, vol. 26, no. 5, pp. 599–613, 2015.

[31] W. Chamlertwat and P. Bhattarakosol, "Discovering Consumer Insight from Twitter via Sentiment Analysis.," *J. Univers. Comput. Sci.*, vol. 18, no. 8, pp. 973–992, 2012.

[32] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? Sentiment Classification using Machine Learning Techniques," *Proc. ACL-02 Conf. Empir. Methods Nat. Lang. Process. - EMNLP '02*, vol. 10, no. July, pp. 79–86, 2002.

[33] G. Li and F. Liu, "Application of a clustering method on sentiment analysis," *J. Inf. Sci.*, vol. 38, no. 2, pp. 127–139, 2012.

[34] A. F. Anta, P. Morere, L. N. Chiroque, and A. Santos, "Sentiment analysis and topic detection of Spanish Tweets: A comparative study of NLP techniques," *Proces. Leng. Nat.*, vol. 50, pp. 45–52, 2013.

[35] P. Ray and A. Chakrabarti, "Twitter Sentiment Analysis for Product Review Using Lexicon Method," *2017 Int. Conf. Data Manag. Anal. Innov. Zeal Educ. Soc. Pune, India, Feb 24-26, 2017*, pp. 211–216, 2017.

[36] E. A. Corrêa, V. Q. Marinho, and L. B. dos Santos, "NILC-USP at SemEval-2017 Task 4: A Multi-view Ensemble for Twitter Sentiment Analysis," pp. 611–615, 2017.

[37] Z. Jianqiang and G. Xiaolin, "Comparison research on text pre-processing methods on twitter sentiment analysis," *IEEE Access*, vol. 5, no. c, pp. 2870–2879, 2017.

[38] C. Silva, “Inductive Inference for large scale text classification,” p. 210, 2008.

[39] V. B. Vaghela, B. M. Jadav, and M. E. Scholar, “Analysis of Various Sentiment Classification Techniques,” *Int. J. Comput. Appl.*, vol. 140, no. 3, pp. 975–8887, 2016.

[40] R. Deerwester, Scott; Dumais, Susan T; Furnas, George W; Landauer, Thomas K; Harshman, “Indexing by latent semantic analysis.” New York, New York, USA, pp. 391–407, 1990.

[41] “Principal Component Analysis and Factor Analysis,” in *Principal Component Analysis*, New York: Springer-Verlag, 2002, pp. 150–166.

[42] A. Sharma and S. Dey, “Performance Investigation of Feature Selection Methods and Sentiment Lexicons for Sentiment Analysis,” *Int. J. Comput. Appl.*, no. June, pp. 15–20, 2012.

[43] J. Ross, Q. Morgan, and K. Publishers, “Book Review : C4 . 5 : Programs for Machine Learning,” vol. 240, pp. 235–240, 1994.

[44] I. Kononenko, “Estimating attributes: Analysis and extensions of RELIEF,” pp. 171–182, 1994.

[45] P. Koncz and J. Paralic, “An approach to feature selection for sentiment analysis,” *2011 15th IEEE Int. Conf. Intell. Eng. Syst.*, pp. 357–362, 2011.

[46] H. Cui, V. Mittal, and M. Datar, “Comparative experiments on sentiment classication for online product reviews,” *Entropy*, pp. 1265–1270, 2003.

[47] S. Dasgupta and V. Ng, “Topic-wise, sentiment-wise, or otherwise?: Identifying the hidden dimension for unsupervised text classification,” *2009 Conf. Empir. Methods Nat. Lang. Process.*, no. August, pp. 580–589, 2009.

[48] K. Dave, K. Dave, S. Lawrence, S. Lawrence, D. M. Pennock, and D. M. Pennock, “Mining the peanut gallery: Opinion extraction and semantic classification of product reviews,” *Proc. 12th Int. Conf. World Wide Web*, pp. 519–528, 2003.

[49] A. Sharma and S. Dey, “A comparative study of feature selection and machine learning techniques for sentiment analysis,” *Proc. 2012 ACM Res. Appl. Comput. Symp. (RACS 2012)*, pp. 1–7, 2012.

[50] J. R. Quinlan, “Induction of Decision Trees,” *Mach. Learn.*, vol. 1, no. 1, pp. 81–106,

1986.

[51] L. M. Rojas-Barahona, “Deep learning for sentiment analysis: A Survey,” *Lang. Linguist. Compass*, vol. 10, no. 12, pp. 701–719, 2016.

[52] G. Betancour, “Las máquinas de soporte vectorial (SVMs),” *Sci. Tech.*, no. 27, pp. 67–72, 2005.

[53] D. Effrosynidis, S. Symeonidis, and A. Arampatzis, “A Comparison of Pre-processing Techniques for Twitter Sentiment Analysis,” vol. 10450, no. September, 2017.

[54] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. J. Miller, “Introduction to WordNet: An On-line Lexical Database,” *Int. J. Lexicogr.*, vol. 3, no. 4, pp. 235–244, Dec. 1990.

[55] S. Mohammad, C. Dunne, and B. Dorr, “Generating high-coverage semantic orientation lexicons from overtly marked words and a thesaurus,” *EMNLP ’09 Proc. 2009 Conf. Empir. Methods Nat. Lang. Process.*, vol. 2, no. August, pp. 599–608, 2009.

[56] L. I. KUNCHEVA, *Combining pattern classifiers: methods and algorithms*, Jhon Wiley. Canada, 2004.

[57] Z.-H. Zhou, *Ensemble Methods: Foundations and Algorithms*. 2012.

[58] M. J.A. Benediktsson, J. Kittler, F. Roli, “Multiple Classifier Systems,” *Springer*, vol. 5519, 2009.

[59] N. F. F. Da Silva, E. R. Hruschka, and E. R. Hruschka, “Tweet sentiment analysis with classifier ensembles,” *Decis. Support Syst.*, vol. 66, pp. 170–179, 2014.

[60] M. Sokolova and G. Lapalme, “A systematic analysis of performance measures for classification tasks,” *Inf. Process. Manag.*, vol. 45, no. 4, pp. 427–437, 2009.

[61] I. Mozetič, L. Torgo, V. Cerqueira, and J. Smailović, “How to evaluate sentiment classifiers for Twitter time-ordered data?,” *PLoS One*, vol. 13, no. 3, pp. 1–20, 2018.

[62] S. Sohangir, D. Wang, A. Pomeranets, and T. M. Khoshgoftaar, “Big Data: Deep Learning for financial sentiment analysis,” *J. Big Data*, vol. 5, no. 1, 2018.

[63] F. H. Khan, U. Qamar, and S. Bashir, “A semi-supervised approach to sentiment analysis using revised sentiment strength based on SentiWordNet,” *Knowl. Inf. Syst.*,

vol. 51, no. 3, pp. 851–872, 2017.

[64] A. S. Hosseini, “Sentence-level emotion mining based on combination of adaptive Meta-level features and sentence syntactic features,” *Eng. Appl. Artif. Intell.*, vol. 65, pp. 361–374, 2017.

[65] T. Mahmood, T. Iqbal, F. Amin, W. Lohanna, and A. Mustafa, “Mining Twitter big data to predict 2013 Pakistan election winner,” in *INMIC*, 2013, pp. 49–54.

[66] D. K. and R. Kunal Keshwani, Piyush Agarwal, “Prediction ofMarketMovement of Gold, Silver and Crude Oil Using Sentiment Analysis,” *Int. J. Sci. Res.*, pp. 101–109, 2017.

[67] H. Krishnan, “Sentiment Analysis of Tweets for Inferring Popularity of Mobile Phones,” *Int. J. Comput. Appl.*, vol. 157, no. 2, pp. 2–4, 2017.

[68] N. Öztürk and S. Ayvaz, “Sentiment analysis on Twitter: A text mining approach to the Syrian refugee crisis,” *Telemat. Informatics*, no. October, 2017.

[69] L. C. Escalante, “El patrón de arquitectura n-capas con orientación al dominio como solución en el diseño de aplicaciones empresariales.,” *Tecnol. Desarro.*, vol. 11, no. 1, pp. 59–66, 2016.

Apéndices

AP1. Detalle del Webservice

- Estructura del Web service

A continuación, se presenta la *Figura 49*, en la cual se detallan los directorios, así como archivos más importantes para el funcionamiento del sistema:

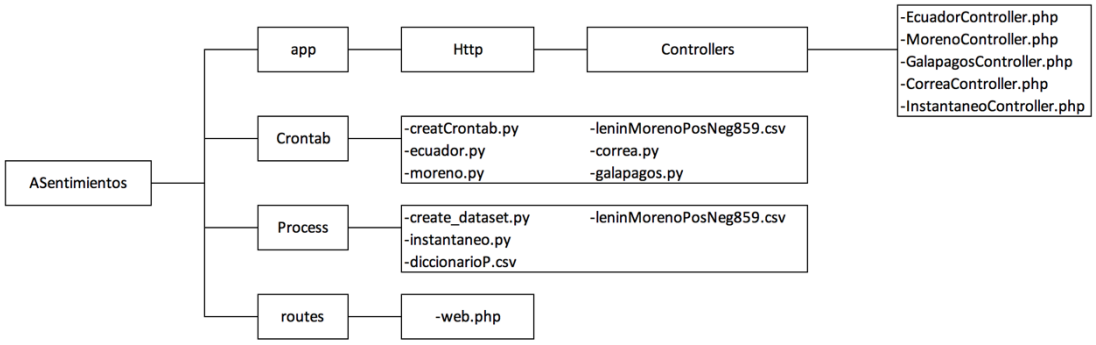

Figura 49. Estructura del servicio web

En el directorio “Controllers” tenemos archivos con formato php, los cuales permiten la comunicación entre la capa de presentación con la cada de datos.

En el directorio “Crontab” tenemos varios archivos que se ejecutaran automáticamente, mediante un proceso crontab, la programación está definida de la siguiente manera:

- Hora 20:00h se ejecuta el archivo ecuador.py
- Hora 21:00h se ejecuta el archivo moreno.py
- Hora 22:00h se ejecuta el archivo galapagos.py
- Hora 23:00h se ejecuta el archivo correa.py

En el directorio “Processs” tenemos los archivos, que permite una consulta online de cualquier tema, en el usuario final ingrese al aplicativo para ser analizado.

En el directorio “routes” tenemos un archivo de configuración, en donde se determinan las URLs disponibles para que la capa de presentación las pueda consumir.

- APIs Rest disponibles en el *webservice*

Dentro del webserice desarrollado, se provee de servicios API Rest que usa directamente en la capa de presentación; a continuación, se los detalla:

Para realizar la *consulta online* se tiene el siguiente API Rest, que se muestra en la *Tabla 28*:

URL	http://uceipl.ddnsking.com/ASentimientos/public/instantaneo/			
Formato de respuesta	**Funcionalidad**	**Parámetro requerido**	**Descripción**	**Tipo**
JSON	Online	Cualquier palabra	Devuelve el número de tweets positivos y negativos	Get

Tabla 28. API Rest para consulta online

Para realizar la consulta offline tema ecuador se tiene el siguiente API Rest, que se muestra en la *Tabla 29*:

URL	http://uceipl.ddnsking.com/ASentimientos/public/ecuador/			
Formato de respuesta	**Funcionalidad**	**Parámetro requerido**	**Descripción**	**Tipo**
JSON	Top List	Intervalo de días	Devuelve el número de tweets positivos y negativos	Get

Tabla 29. API Rest para consultar el tema "Ecuador"

Para la consulta offline tema moreno se tiene el siguiente API Rest, que se muestra en la *Tabla 30*:

URL	http://uceipl.ddnsking.com/ASentimientos/public/moreno/			
Formato de respuesta	**Funcionalidad**	**Parámetro requerido**	**Descripción**	**Tipo**
JSON	Top List	Intervalo de días	Devuelve el número de tweets positivos y negativos	Get

Tabla 30. API Rest para consultar el tema "Moreno"

Para la consulta offline tema galápagos se tiene el siguiente API Rest, que se muestra en la *Tabla 31*:

URL	http://uceipl.ddnsking.com/ASentimientos/public/galapagos/			
Formato de respuesta	**Funcionalidad**	**Parámetro requerido**	**Descripción**	**Tipo**
JSON	Top List	Intervalo de días	Devuelve el número de tweets positivos y negativos	Get

Tabla 31. API Rest para consultar el tema "Galápagos"

Para la consulta offline tema correa se tiene el siguiente API Rest, que se muestra en la *Tabla 32*.

URL	http://uceipl.ddnsking.com/ASentimientos/public/correa/			
Formato de respuesta	**Funcionalidad**	**Parámetro requerido**	**Descripción**	**Tipo**
JSON	Top List	Intervalo de días	Devuelve el número de tweets positivos y negativos	Get

Tabla 32. API Rest para consultar el tema "Correa"

Glosario

Corpus	Es una gran colección de textos. Es un cuerpo de material escrito o hablado sobre el cual se basa un análisis lingüístico.
Microblogging	Es un medio de difusión que existe en forma de blogging. Un microblog difiere de un blog tradicional en que su contenido suele ser más pequeño en tamaño de archivo tanto real como agregado. Los microblogs "permiten a los usuarios intercambiar pequeños elementos de contenido como oraciones cortas, imágenes individuales o enlaces de video.
Tesauro	Es una lista de palabras con significados similares sinónimos, habitualmente acompañada por otra lista de antónimos.
WordNet	Es una base de datos léxica del Idioma inglés que agrupa palabras en inglés en conjuntos de sinónimos llamados synsets, proporcionando definiciones cortas y generales y almacenando las relaciones semánticas entre los conjuntos de sinónimos.
Twimemachine	Es una aplicación web que te va a permitir recuperar tus tweets publicados con mayor antigüedad a los 3200, que como se sabe es el límite que impone Twitter para mostrar tweets.