Máster Universitario en

Ingeniería **C**omputacional y **S**istemas **I**nteligentes

Universidad del País Vasco
Euskal Herriko Unibertsitatea

Konputazio Zientziak eta Adimen Artifiziala Saila –
Departamento de Ciencias de la Computación e Inteligencia Artificial

Tesis de Máster

Oscar Miguel Cumbicus Pineda

CATEGORIZACIÓN AUTOMÁTICA DE TWEETS SOBRE EL TEMA POLÍTICO ELECTORAL APLICANDO ALGORITMOS DE CLASIFICACIÓN SUPERVISADA

Tutor

Basilio Sierra Araujo

Departamento de Ciencia de la Computación e Inteligencia Artificial

Facultad de Informática, UPV/EHU

Julio 2017

K I S A

I C S

informatika fakultatea
facultad de informática

KZAA

ÍNDICE

RESUMEN 4
INTRODUCCIÓN 4
1. ESTADO DEL ARTE 6
1.1. Minería de datos 6
1.2. Aprendizaje Automático en la Clasificación de Texto 7
1.3. Red Social Twitter 8
1.4. Software WEKA 9
1.5. Algoritmos de aprendizaje supervisado. 10
1.5.1. Validación de los clasificadores 10
1.5.2. Naive Bayes (NB) 12
1.5.3. Logistic Regression 12
1.5.4. Máquinas de Soporte Vectorial (SMV) 13
1.5.5. K-vecinos más cercanos 14
1.5.6. Tablas de Decisión 15
1.5.7. Árboles de Clasificación 16
1.5.8. Clasificador Random Forest 18
1.6 SMOTE (Syntetic Minority Over-Sampling Technique) 18
2. METODOLOGÍA 19
2.1. Elecciones Presidenciales 2017 en Ecuador 21
2.2. Fase 1: Recolección de tweets 24
2.3. Fase 2: Obtención del Bag of Words 25
2.4. Fase 3: Equilibrar las clases con SMOTE 26
2.5. Fase 4: Ejecución de los clasificadores 28
3. EVALUACIÓN Y RESULTADOS 29
3.1. Evaluación 29

3.2. Resultados 29
3.3. Análisis de resultados 33
4. TRABAJOS FUTUROS 36
5. CONCLUSIONES 37
6. BIBLIOGRAFÍA 37

CATEGORIZACIÓN AUTOMÁTICA DE TWEETS SOBRE EL TEMA POLÍTICO ELECTORAL APLICANDO ALGORITMOS DE CLASIFICACIÓN

RESUMEN

Actualmente el incremento del uso de redes sociales para compartir contenidos y opiniones de diferentes índoles permite tener un gran volumen de información. Twitter, es una de las redes sociales más utilizadas por su fácil acceso y manejo. Los usuarios de esta red se convierten no solamente en actores pasivos de recepción y consumo de información, sino también en generadores de contenidos. El análisis de tweets requiere de un proceso sistemático para su recolección, transformación y clasificación, es por ello que en esta tesis de master se describe un trabajo de investigación que aplica técnicas de minería de datos para la obtención de clasificadores que permitan identificar automáticamente la categoría (Positiva, Negativa, Neutral) de la opinión pública manifestada en Twitter correspondientes a temas políticos.

Atendiendo a estas consideraciones se utilizó un total de 745 tweets recolectados en español de las principales cuentas de medios de comunicación, personajes políticos y organizaciones políticas. Estos tweets fueron preprocesados, transformados y los resultados obtenidos de la aplicación de los algoritmos de clasificación son presentados, analizados y comparados.

Palabras clave: Clasificación supervisada, categorización de tweets, algoritmos, twitter y política.

INTRODUCCIÓN

Hoy en día es indiscutible el papel que están jugando las redes sociales en Internet, ya que su incidencia sobre la vida política, económica y social, permitieron transformar la dinámica para comunicarnos y adquirir información (Gálvez Pérez, 2015).

Twitter es en la actualidad uno de los grandes protagonistas de la red global. Esta plataforma de comunicación ha establecido, a base de popularidad, una nueva forma de comunicación: el microblogging. Luego de más de 8 años de uso y más de 500 millones de usuarios, Twitter se ha convertido en una plataforma esencial para el seguimiento, difusión y coordinación de eventos de diversa naturaleza e importancia.

El uso de los medios sociales –y, en particular, el microblogging de Twitter– han transformado la comunicación en política rutinaria. En las redes sociales han surgido nuevos modos de comunicación que incluyen la discusión ciudadana de asuntos políticos. Los mensajes, controlados

por poderosos gatekeepers en los medios tradicionales, fluyen ahora libremente en un medio pasivo en el que los comunicantes son los propios ciudadanos y los mensajes están diversificados. (Moya Sánchez & Herrera Damas, 2015)

Una democracia representativa de ciudadanos instruidos, requiere de una modalidad de comunicación en la que aquellos puedan trasladar su opinión a los gobernantes como reacción a las políticas aplicadas o que pretenden implementar. Las TICs crean los espacios para una comunicación política diferente, en la que todos los actores –ciudadanos, dirigentes políticos, periodistas y otros influyentes– pueden intervenir en igualdad de condiciones. La interacción es clave en el nuevo proceso de comunicación que proponen los medios sociales. (Moya Sánchez & Herrera Damas, 2015)

El Consejo Nacional Electoral de la República del Ecuador (CNE), es el máximo organismo de sufragio. Sus funciones son organizar, dirigir, vigilar y garantizar de manera trasparente y eficaz los procesos electorales, organizar procesos de referéndum, consulta popular o revocatorias de mandato, mantener el registro permanente de organizaciones políticas y velar por el desarrollo transparente de las elecciones (Consejo Nacional Electoral, 2016)**.**

El Consejo Nacional Electoral ha definido objetivos y ejes estratégicos que responden a la consecución de los principios constitucionales de calidad y eficiencia en el servicio público. Por ello es importante identificar el posicionamiento de la institución dentro del Estado y basándonos en el objetivo "Incrementar la eficacia y eficiencia institucional para brindar servicios de calidad" y su eje estratégico "Fortalecimiento Institucional" se realizan mediciones en las redes sociales de la imagen institucional. El CNE tiene 24 delegaciones Provinciales, una por cada provincia, para desconcentrar los servicios electorales en todo el país.

La misión del Consejo Nacional Electoral es fortalecer la democracia en el Ecuador, garantizando los derechos políticos y la organización política de la ciudadanía, promoviendo el ejercicio de la democracia comunitaria y ejerciendo rectoría, planificación, regulación y el control de los mecanismos de democracia directa y representativa (Consejo Nacional Electoral, 2016)**.**

El 18 de febrero de 2016, a un año de las elecciones, el CNE aprobó el cronograma electoral en el que la primera vuelta electoral tendría lugar el 19 de febrero de 2017 donde se elegiría al nuevo presidente y vicepresidente del Ecuador para el periodo 2017-2021; también se realizaría la elección de cinco representantes al Parlamento Andino y 137 Asambleístas para el mismo periodo, además de una consulta popular sobre la opinión de los ecuatorianos en el tema de los funcionarios públicos que tengan cuentas y empresas en paraísos fiscales.

Fueron habilitados 12.816.698 electores en todo el Ecuador para participar de este proceso electoral, donde participaron también 7 partidos nacionales, 9 movimientos nacionales y 54 movimientos políticos provinciales, dando un total de 70 organizaciones políticas.

Ocho binomios fueron presentados para la dignidad de presidente y vicepresidente; debido a que ninguno de los binomios presidenciales participantes en este proceso electoral logró obtener la votación para ser elegido en una sola vuelta, se tuvo que ejecutar una segunda vuelta electoral con los dos binomios presidenciales más votados. Esté nuevo proceso electoral se realizó el día 2 de abril de 2017, resultando ganador el binomio presidencial del Lic. Lenin Moreno y el Ing. Jorge Glas representantes del movimiento Alianza PAIS, con un votación igual 51.16% del universo de sufragantes.

El presente trabajo de investigación se enfoca en la categorización de los tweets recolectados principalmente de cuentas de medios de comunicación, personajes políticos y organizaciones políticas a través del equipo de comunicación de la delegación provincial electoral de Loja; los tweets fueron recolectados de esas cuentas a quienes realizaron publicaciones relacionadas con el Consejo Nacional Electoral, tweets que se clasificaron en tres categorías (clases) denominadas: positiva, negativa y neutral; las cuales se definieron a priori en cada Tweet.

La importancia del presente trabajo radica en encontrar el mejor modelo para la categorización automática de tweets, aplicando algoritmos de clasificación supervisada; lo que minimizará significativamente el trabajo actual que realiza el área de comunicación de la delegación provincial electoral de Loja.

1. ESTADO DEL ARTE

Esta sección tiene como objetivo presentar la base teórica sobre la que se fundamenta el desarrollo del trabajo: minería de datos, aprendizaje automático, redes sociales y los principales algoritmos de clasificación.

1.1. Minería de datos

Los textos albergan una gran cantidad de información que los ordenadores no pueden analizar, ya que tradicionalmente han sido tratados como simples secuencias de caracteres. Es necesario aplicar métodos y algoritmos para procesar estos textos y extraer información de utilidad para ellos. Este campo se encuentra en constante crecimiento debido a la proliferación de las redes sociales ya que se estima que alrededor del 80% de la información relacionada con el mundo empresarial se encuentra almacenada en forma de texto (Amaya de la Peña, 2015).

Existen diferentes técnicas que posibilitan la explotación de los datos, extrayendo información que no es detectada a simple vista. Una de estas técnicas es la denominada Minería de Datos, la cual combina técnicas semiautomáticas de inteligencia artificial, análisis estadístico, bases de datos y visualización gráfica, para la obtención de información que no esté representada explícitamente en los datos. La Minería de Datos descubre relaciones, tendencias, desviaciones, comportamientos atípicos, patrones y trayectorias ocultas, con el propósito de soportar los procesos de toma de decisiones con mayor conocimiento. La Minería de Datos se puede ubicar en el nivel más alto de la evolución de los procesos tecnológicos de análisis de datos (Martínez, 2001).

Hemos utilizado la minería de datos para verificar los patrones que tienen los tweets recolectados y poderlos clasificar en su respectiva clase.

1.2. Aprendizaje Automático en la Clasificación de Texto

Hasta la década de los 80 el paradigma de la clasificación de texto era abordado desde el enfoque de la Ingeniería del Conocimiento. Se construían sistemas expertos en los que se trataba de plasmar, en forma de reglas, el conocimiento de un experto. Estos sistemas son estáticos y dependientes del dominio, ya que en todo momento están condicionados al factor humano para su desarrollo. Posteriormente se introdujeron nuevas técnicas de aprendizaje automático, aunque también pueden incluir conocimiento de fondo como WordNet o basarse en los contenidos léxico-semántico comunes entre textos. En cualquier caso se elimina el pesado trabajo de la extracción manual de conocimiento que llevaban a cabo los ingenieros del conocimiento y las técnicas se hacen independientes de la temática que se desea tratar (Echegoyen, 2007).

Pueden distinguirse dos grandes grupos a la hora de estudiar la clasificación de texto desde el punto de vista del aprendizaje automático. En primer lugar una clasificación supervisada, dónde tomando como base un conjunto de documentos bien clasificados, se construye un modelo (clasificador φ) para tratar de predecir la categoría de nuevos textos. Por otro lado, una clasificación no supervisada, donde no existe información previa y se trata de descubrir la existencia de clases en los documentos. En este contexto aparece la tarea denominada Text Clustering, que trata de distinguir distintas categorías en un conjunto de documentos y agruparlos en torno a ellas. Esto puede resultar muy interesante en ámbitos como internet donde existen grandes volúmenes de textos sin una jerarquía específica y su agrupamiento y clasificación resulta muy útil (Echegoyen, 2007).

Efectivamente dentro de este estudio se ha realizado una clasificación supervisada de una base de datos de tweets que primeramente fueron clasificados en forma manual y luego se construyeron siete modelos que predijeron la clase a la que los nuevos tweets deberían pertenecer.

1.3. Red Social Twitter

Actualmente un gran porcentaje de la información que se genera en Internet, parte de la llamada Web 2.0. Este término se usa para designar aquellas páginas que prestan especial atención a los usuarios, ya que éstos son considerados los principales creadores de contenido. Estas páginas centran sus esfuerzos en proporcionar un fácil acceso e intentan que la comunicación e interacción entre ellos sea lo más fluida y sencilla posible. Algunos ejemplos de este fenómeno son los blogs, YouTube, Facebook o Twitter (Amaya de la Peña, 2015).

De entre todas las redes sociales Twitter es la que adquirido una mayor relevancia en el panorama político actual. Sus principales características son la brevedad y la rapidez a la hora de escribir mensajes, lo que favorece que se publique un gran número de mensajes al día. Twitter se ha convertido en un espacio online de debate alrededor de multitud de temas. Su estructura horizontal provoca que los usuarios se sientan más cercanos a los políticos y famosos, lo cual favorece la comunicación entre ambas partes (Amaya de la Peña, 2015).

En el Ecuador la utilización de redes sociales en los últimos años se ha intensificado; es por ello que para las elecciones presidenciales de 2017 se pudo observar cómo las organizaciones políticas realizaron estrategias comunicacionales con esta red social y otras, para llegar a estratos de la sociedad a la que comúnmente no han llegado, como por ejemplo los jóvenes. Esas estrategias comunicacionales también incluyen el realizar pronunciamientos sobre el actuar del CNE ya que es el ente rector de los procesos electorales. Para realizar estos comentarios se utilizaron los llamados *hashtags*, que son etiquetas que permiten identificar fácilmente tweets sobre un tema o noticia determinada. Los usuarios no tienen nada más que incluir en sus tweets el hashtag precedido del símbolo #, para el caso en estudio algunos de los hashtags utilizados fueron: #Elecciones2017EC, #Elecciones19F, #CNE, #EcuadorDecide, #SimulacroNacionalCNE y también se hicieron mención a algunas cuentas de institucionales como: @cnegobec, @CNELoja, @JuanPabloPozoBa, @OscarCumbicus (Figura 1), de esta forma utilizando estos elementos como base dentro de twitter, se pudieron recoger comentarios relacionados con el Consejo Nacional Electoral y de la Delegación Provincial Electoral de Loja.

Figura 1. Ejemplo de tweet candidatos presidenciales de Ecuador 2017 (Política en Tweets, 2017)

1.4. Software WEKA

WEKA (Waikato Environment for Knowledge Analysis) es un software gratuito de aprendizaje automático y minería de datos distribuido bajo licencia GNU-GPL. Es una plataforma desarrollada en Java por la Universidad de Waikato, en Nueva Zelanda (Lage García, 2014).

Weka tiene una gran colección de algoritmos para tareas de minería de datos y modelado predictivo. Éstos pueden aplicarse directamente a un conjunto de datos a través de su interfaz gráfica o bien pueden ser llamados desde un programa externo a través de las API proporcionadas.

En este trabajo se ha tomado la primera opción, concretamente se ha utilizado la interfaz Explorer. Weka también incluye herramientas para el preprocesamiento de los datos (filtros), clasificación (árboles, tablas), clustering, reglas de asociación, y adicionalmente, diversas formas de visualización de los datos, tanto en el inicio del proceso de carga de datos, como después de haber aplicado un algoritmo.

En este proyecto WEKA ha sido utilizado para generar un modelo que determina la clase de un tweet, es decir, lo clasificará como positivo, negativo, neutro. Para poder entrenar y testear este modelo del clasificador se utiliza el corpus detallado en la sección 2.1 de este documento.

1.5. Algoritmos de aprendizaje supervisado.

1.5.1. Validación de los clasificadores

Un test de validación nos permitire afinar los parámetros del algoritmo para obtener el resultado esperado. En este caso el test de validación no es más que una serie de muestras ya clasificadas previamente en las que se prueba el modelo.

Una forma de estimar la precisión de un modelo es coger los datos de muestra y hacer dos subconjuntos: uno seria los datos de entrenamiento y otro los datos de prueba. Los datos de entrenamiento se utilizarán para entrenar el modelo y los de prueba para validarlo. Entrenamos el algoritmo y luego probamos el modelo sobre los datos de prueba, que al estar ya etiquetados podemos saber la precisión del modelo.

Existen dos tipos de validación cruzada para la estimación de exactitud de clasificación:

- K-fold cross-validation
- Leave-one-out cross-validation.

En la validación **cruzada de K iteraciones o K-fold cross-validation** (Figura 2), los datos de muestra se dividen en K subconjuntos. Uno de los subconjuntos se utiliza como datos de prueba y el resto (K-1) como datos de entrenamiento. El proceso de validación cruzada es repetido durante k iteraciones, con cada uno de los posibles subconjuntos de datos de prueba. Finalmente se realiza la media aritmética de los resultados de cada iteración para obtener un único resultado. Este método es muy preciso puesto que evaluamos a partir de K combinaciones de datos de entrenamiento y de prueba, pero aun así tiene una desventaja, y es que, es lento desde el punto de vista computacional.

En la práctica, la elección del número de iteraciones depende de la medida del conjunto de datos. Para la presente tesis utilizamos la validación cruzada de 10 iteraciones (10-fold cross-validation).

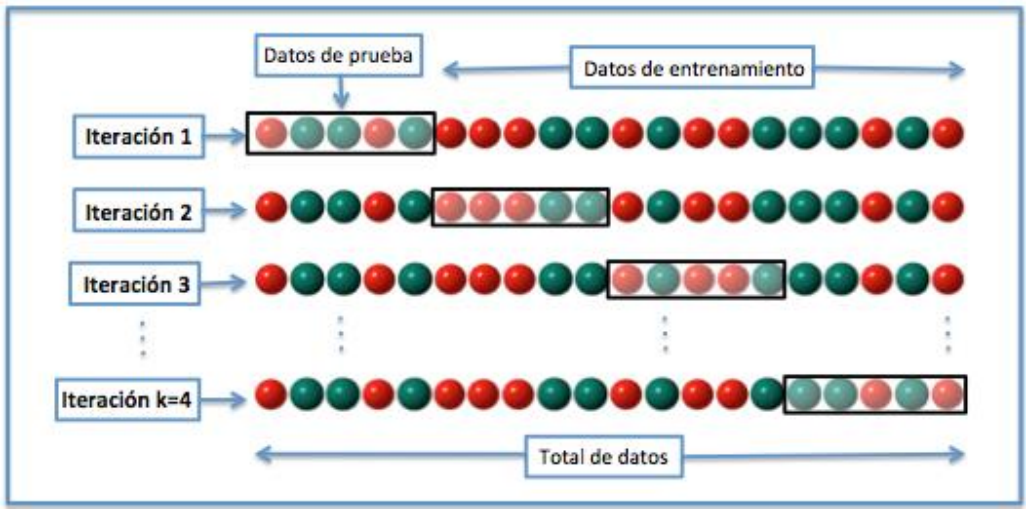

Figura 2. Validación cruzada de K iteraciones con K=4. (Joanneum, 2005-2006)

La validación **cruzada dejando uno fuera o Leave-one-out cross-validation (LOOCV)** (Figura 3), implica separar los datos de forma que para cada iteración tengamos una sola muestra para los datos de prueba y todo el resto conformando los datos de entrenamiento. La evaluación viene dada por el error, y en este tipo de validación cruzada el error es muy bajo, pero en cambio, a nivel computacional es muy costoso, puesto que se tienen que realizar un elevado número de iteraciones, tantas como N muestras tengamos y para cada una analizar los datos, tanto de entrenamiento como de prueba.

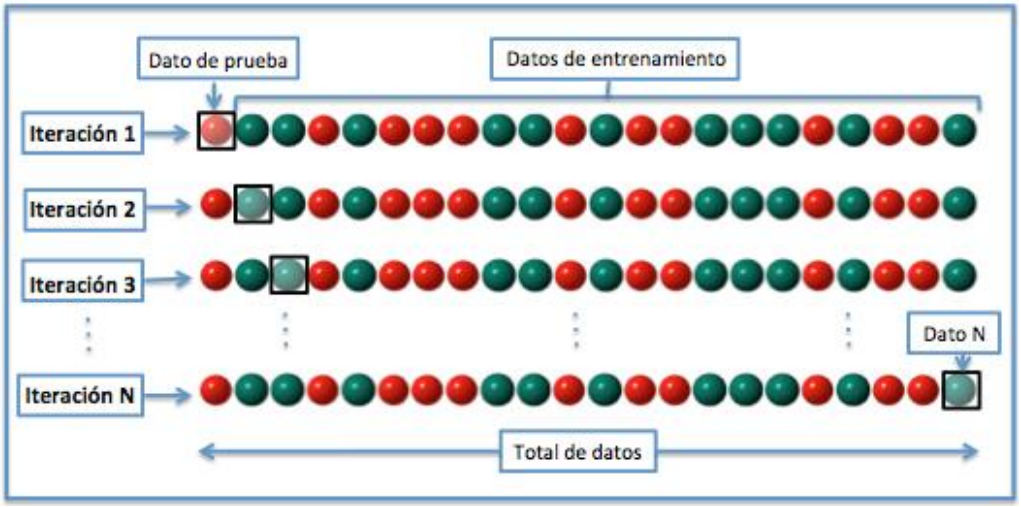

Figura 3. Validación cruzada dejando uno fuera (LOOCV) (Joanneum, 2005-2006)

1.5.2. Naive Bayes (NB)

El aprendizaje Bayesiano se basa en la asunción de que las cantidades de interés están gobernadas por distribuciones de probabilidades y que las decisiones óptimas se pueden hacer razonando sobre estas probabilidades y sobre los datos observados (Buill Vilches, 2014)**.**

El clasificador Naive Bayes es un clasificador probabilístico que se basa en aplicar el teorema de Bayes; NB asume la independencia de las variables predictoras dado el valor de la clase; es por eso que se le llama clasificador bayesiano ingenuo (naive). (Martis Cáceres, 2012)

Como se explica en (Tomás Diaz, 2010)**,** desde el punto de vista de la clasificación de textos se puede decir que se asume la independencia de las palabras, es decir, la probabilidad condicional de una palabra dada una clase se asume que es independiente de la probabilidad condicional de otras palabras dada esa clase.

Definiendo más formalmente, sea {1…K} el conjunto de clases posibles y {xi,1, … , xi,m}, el conjunto de valores de las características del ejemplo xi, el algoritmo Naive Bayes selecciona la clase que maximiza P(k|$x_{i,1},\ldots,x_{i,m}$): (Martis Cáceres, 2012)

$$\arg\max_{x} P(k|x_{i,1},\ldots,x_{i,m}) \approx \arg\max_{x} P(k)\prod_j P(x_{i,j}|k) \qquad (2)$$

Fuente: (Martis Cáceres, 2012)

Las probabilidades P(k) y P ($x_{i,j}$,K), se estiman a partir del corpus de aprendizaje a través de las frecuencias relativas. Es conocido que el algoritmo de Naive Bayes tiene una limitante para trabajar en espacios de gran dimensionalidad, en otros términos no puede trabajar con un gran número de características de aprendizaje (Neyra, 2016)**.**

1.5.3. Logistic Regression

El clasificador de Logistic Regresion aprende funciones de probabilidad de la forma P(Y| X), donde Y es la clase de variable y X el vector de atributos (Hosmer Jr, 2013).

Logistic Regresion supone una función paramétrica de la distribución de P(Y|X), y basado en los datos del entrenamiento estima los parámetros de la distribución. La distribución es generalmente una función logística, por lo tanto justificando su nombre así como el rango de probabilidades entre 0 y 1. P(Y| X) puede ser una combinación lineal del vector de atributo de predicción (Masías, y otros, 2016).

Batista y Ribeiro (Batista & Ribeiro, 2012), proponen un sistema de clasificación basado en modelos de regresión logística. Mediante clasificadores de máxima entropía para eventos independientes, crean un clasificador binario para cada tópico que estima la probabilidad de que un tuit pertenezca o no a dicha categoría. Estos clasificadores toman como atributos de entrada los unigramas y bigramas de palabras para cada mensaje, así como ciertas características propias de la jerga de Twitter, seleccionando el tópico más probable en función del valor de confianza obtenido para cada clasificador (Calvo Vilares, 2014).

1.5.4. Máquinas de Soporte Vectorial (SMV)

Dado un conjunto de ejemplos de entrenamiento, una SVM los representa como puntos en el espacio, separando las clases por la mayor distancia posible. Cuando las nuevas muestras se ponen en correspondencia con dicho modelo, según su proximidad serán asignadas a una u otra clase. Es decir, si un punto nuevo pertenece a una categoría o la otra (Blanco & Hermida, 2016).

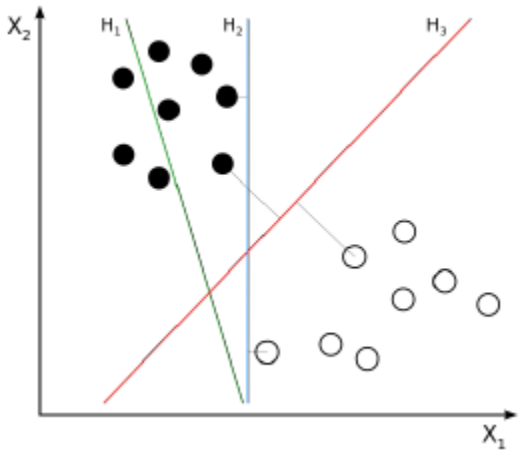

Figura 4. H1 no separa las clases. H2 si pero H3 lo hace con la separación máxima (Blanco & Hermida, 2016)

Existen otros enfoques como el propuesto por Pla y Hurtado (Pla & Hurtado, 2013): en su estudio presentan una cascada de clasificadores binarios SMO, pertenecientes a la familia de los SVM (Support Vector Machine), que son entrenados para cada tópico y posteriormente utilizados para clasificar un conjunto de tweets desconocidos. La cascada de clasificadores asigna a cada tweet las temáticas detectadas por al menos un clasificador. Dado que cabe la posibilidad de que ningún tópico sea predicho, los autores emplean una segunda cascada de clasificadores binarios libSVM (Chang & Lin, 2011), a modo de back-off. Como resultado, cada clasificador devuelve un factor de probabilidad que indica la confianza con la que un tweet puede ser asignado a una categoría. En este caso, los autores seleccionan el tópico para el que se obtiene un mayor factor confianza, descartando todos los demás. La efectividad de los clasificadores SVM ha sido estudiada por otros

autores, sin considerar una arquitectura en cascada. Martínez-Cámara (Martínez Cámara, García Cumbreras, Martín Valdivia, & Ureña López, 2013) emplean un clasificador SVM entrenado con una bolsa de términos constituida por palabras extraídas del corpus, así como una colección de hashtags y palabras extraída de Google AdWords KeyWordTool (Calvo Vilares, 2014).

1.5.5. K-vecinos más cercanos

Es uno de los algoritmos más sencillos de implementar. La idea en la que se basa reside en comparar un documento con sus k vecinos, de modo que este documento será clasificado del tipo del que haya más documentos entre esos k vecinos. Se trata de un algoritmo sencillo y eficaz especialmente cuando existen un gran número de categorías diferentes. El parámetro k deberá ser decidido a priori, y en función de la decisión que se tome se obtendrán unos resultados u otros (Godino Martínez, 2014).

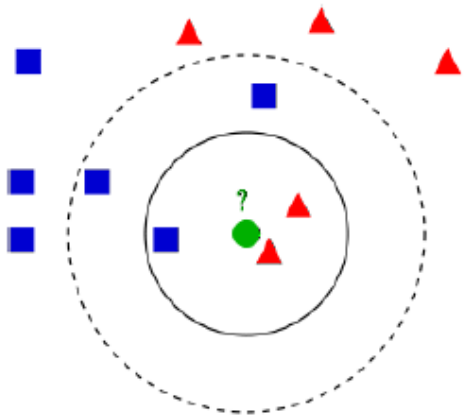

Figura 5. Algoritmo KNN, K=3 Vs Algoritmo KNN, K=5 (Godino Martínez, 2014)

En la figura 5 se comprueba como la elección de dicho parámetro influye de manera notoria en el resultado del algoritmo. Si k = 3, el punto a clasificar en el caso de la figura 5 (el círculo verde), se clasifica como triángulo rojo puesto que hay dos de este tipo por tan solo uno del grupo de cuadrados azules; no obstante, si k = 5 el círculo verde se clasifica como cuadrado azul puesto que de entre sus cinco vecinos más cercanos existen tres elementos de este tipo por tan solo dos de los triángulos (Godino Martínez, 2014).

1.5.6. Tablas de Decisión.

La tabla de decisión constituye la forma más simple y rudimentaria de representar la salida de un algoritmo de aprendizaje, que es justamente representarlo como la entrada. Estos algoritmos consisten en seleccionar subconjuntos de atributos y calcular su precisión para predecir o clasificar los ejemplos. Una vez seleccionado el mejor de los subconjuntos, la tabla de decisión estará formada por los atributos seleccionados (más la clase), en la que se insertarán todos los ejemplos de entrenamiento únicamente con el subconjunto de atributos elegido. Si hay dos ejemplos con exactamente los mismos pares atributo-valor para todos los atributos del subconjunto, la clase que se elija será la media de los ejemplos (en el caso de una clase numérica) o la que mayor probabilidad de aparición tenga (en el caso de una clase simbólica). La precisión de un subconjunto S de atributos para todos los ejemplos de entrenamientos se calculará mediante la ecuación:

$$\boldsymbol{precisión}(\boldsymbol{S}) = \frac{\boldsymbol{ejemplos\ bien\ clasificados}}{\boldsymbol{ejemplos\ totales}}$$

El algoritmo en weka consiste en ir seleccionando uno a uno los subconjuntos, añadiendo a cada uno de los ya probados cada uno de los atributos que aún no pertenecen a él. Se prueba la precisión del subconjunto, bien mediante validación cruzada o leave-one-out y, si es mejor, se continúa con él. Se continúa así hasta que se alcanza maxStale. Para ello, una variable comienza siendo 0 y aumenta su valor en una unidad cuando a un subconjunto no se le puede añadir ningún atributo para mejorarlo, volviendo a 0 si se añade un nuevo atributo a un subconjunto. En cuanto al proceso leave-one-out, es un método de estimación del error. Es una validación cruzada en la que el número de conjuntos es igual al número de ejemplos de entrenamiento. Cada vez se elimina un ejemplo del conjunto de entrenamiento y se entrena con el resto. Se juzgará el acierto del sistema con el resto de instancias según se acierte o se falle en la predicción del ejemplo que se eliminó. El resultado de las n pruebas (siendo n el número inicial de ejemplos de entrenamiento) se promedia y dicha media será el error estimado. Por último, para clasificar un ejemplo pueden ocurrir dos cosas. En primer lugar, que el ejemplo corresponda exactamente con una de las reglas de la tabla de decisión, en cuyo caso se devolverá la clase de dicha regla. Si no se corresponde con ninguna regla, se puede utilizar IBk (equivalente a k-vecinos más cercanos en weka, si se seleccionó dicha opción), para predecir la clase, o la media o moda (López & Herrero, 2006).

1.5.7. Árboles de Clasificación

Un árbol de clasificación (Figura 6), es una forma de representar el conocimiento obtenido en el proceso de aprendizaje inductivo. Puede verse como la estructura resultante de la partición recursiva del espacio de representación a partir del conjunto (numeroso) de prototipos. Esta partición recursiva se traduce en una organización jerárquica del espacio de representación que puede modelarse mediante una estructura de tipo árbol. Cada nodo interior contiene una pregunta sobre un atributo concreto (con un hijo por cada posible respuesta) y cada nodo hoja se refiere a una decisión (clasificación) (González Rubio, 2015).

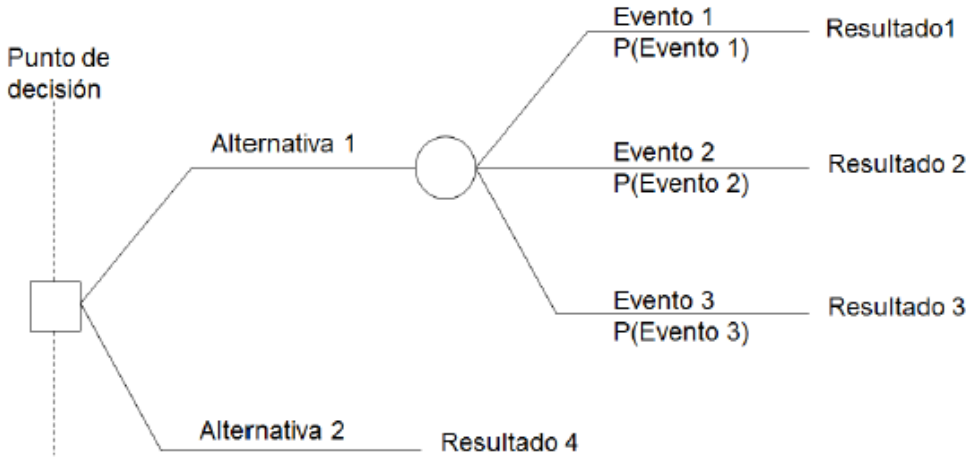

Figura 6. Ejemplo de árbol de decisión Fuente: (González Rubio, 2015)

Para este estudio se ha utilizado en weka el algoritmo de análisis conocido como Árbol de decisión J48. Los árboles de decisión entran dentro de los métodos de clasificación supervisada, es decir, se tiene una variable dependiente o clase, y el objetivo del clasificador es determinar el valor de dicha clase para casos nuevos. El proceso de construcción del árbol comienza por el nodo raíz, el que tiene asociados todos los ejemplos o casos de entrenamiento. Lo primero es seleccionar la variable o atributo a partir de la cual se va a dividir la muestra de entrenamiento original (nodo raíz), buscando que en los subconjuntos generados haya una mínima variabilidad respecto a la clase. Este proceso es recursivo, es decir, una vez que se haya determinado la variable con la que se obtiene la mayor homogeneidad respecto a la clase en los nodos hijos, se vuelve a realizar el análisis para cada uno de los nodos hijos. Aunque en el límite este proceso se detendría cuando todos los nodos hojas contuvieran casos de una misma clase, no siempre se desea llegar a este extremo, para lo cual se implementan métodos de pre-poda y post-poda de los árboles (Taborda, García Gelvez, & Rozo Palacios, 2016).

Por tal motivo en el J48 de weka el parámetro más importante que deberemos tener en cuenta es el factor de confianza para la poda "confidence factor", que influye en el tamaño y capacidad de predicción del árbol construido. Para cada operación de poda, define la probabilidad de error que se permite a la hipótesis de que el empeoramiento debido a esta operación es significativo. A probabilidad menor, se exigirá que la diferencia en los errores de predicción antes y después de podar sea más significativa para no podar. El valor por defecto es del 25%. Según baje este valor, se permiten más operaciones de poda (Jiménez & Álvarez Sierra, 2010).

La configuración por defecto de J48 en weka, fue utilizada en este trabajo con un confidenceFactor=0.25 (Figura 7).

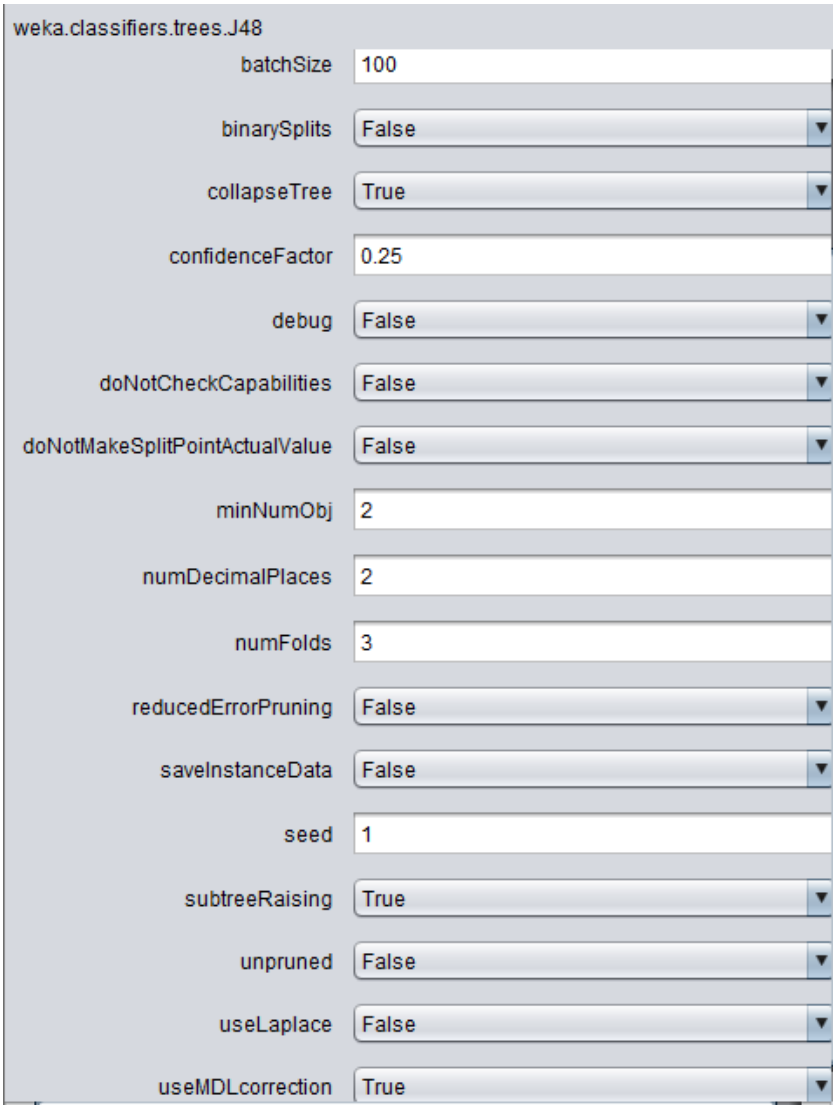

Figura 7. Parámetros utilizados en J48

1.5.8. Clasificador Random Forest

Se basan en el desarrollo de muchos árboles de clasificación. Para clasificar un nuevo objeto desde un vector de entrada, ponemos dicho vector bajo cada uno de los árboles del bosque. Cada árbol genera una clasificación, en términos coloquiales diríamos que cada árbol vota por una clase. El bosque escoge la clasificación teniendo en cuenta la clase más votada sobre todas las del bosque. Cada árbol se desarrolla como sigue (Soltero Domingo & Bodas Sagi, 2012):

- Si el número de casos en el conjunto de entrenamiento es N, prueba N casos aleatoriamente, pero con sustitución, de los datos originales. Este será el conjunto de entrenamiento para el desarrollo del árbol.
- Si hay M variables de entrada, un número m<<M es especificado para cada nodo, m variables son seleccionadas aleatoriamente del conjunto M y la mejor partición de este m es usada para dividir el nodo. El valor de m se mantiene constante durante el crecimiento del bosque.
- Cada árbol crece de la forma más extensa posible, sin ningún tipo de poda.

Una de las características de Random Forest es que se puede calcular la fuerza o importancia usando el método Out–of–Bag (OOB), que mejora la comprensión de los atributos que tienen mayor poder predictivo. El único parámetro que ha de ser elegido es n, el número de variables seleccionadas al azar en cada caso de las variables de N disponibles. El valor de n se determina experimentalmente seleccionando el valor que minimiza la tasa de error para los datos OOB (Masías, y otros, 2016).

1.6 SMOTE (Syntetic Minority Over-Sampling Technique)

Fix y Hodges publicaron el algoritmo de la regla del vecino más cercano. La idea básica del algoritmo es suponer que instancias próximas entre sí tienen mayor probabilidad de pertenecen a la misma clase. Para clasificar una nueva instancia, se realiza un cálculo de la distancia entre cada atributo de la nueva instancia y el resto de instancias del conjunto de datos y se asocia a la clase de la instancia más cercana. El principal inconveniente del algoritmo es el alto coste computacional que tiene. SMOTE (Syntetic Minority Over-sampling Technique) es un algoritmo de oversampling que genera instancias "sintéticas" o artificiales para equilibrar la muestra de datos basado en la regla del vecino más cercano. La generación se realiza extrapolando nuevas instancias en lugar de duplicarlas como hace el algoritmo de Resampling. Para cada una de las instancias minoritarias se buscan las instancias minoritarias vecinas (más cercanas) y se crean N instancias entre la línea que une la instancia original y cada una de las vecinas. El valor de N depende del tamaño de oversampling deseado. Para un caso del 200% por cada instancia de la clase minoritaria deben crearse dos nuevas instancias genéricas. SMOTE es un algoritmo de sobre-muestreo de

ejemplos utilizado para la clase minoritaria (Moreno, Rodríguez, Sicilia, Riquelme, & Ruiz, 2009):

- Crea ejemplos sintéticos en lugar de hacer un sobre-muestreo con reemplazo.
- Opera en el espacio de atributos feature space, en lugar del espacio de datos data space.
- Crea un ejemplo sintético a lo largo de los segmentos de línea que une alguno o todos los k vecinos más cercanos de la clase minoritaria.
- Se eligen algunos de los k vecinos más cercanos de manera aleatoria (no se utilizan todos).
- SMOTE utiliza típicamente k = 5.

El algoritmo de SMOTE realiza los siguientes pasos:

- Recibe como parámetro el porcentaje de ejemplos a sobre-muestrear.
- Calcula el número de ejemplos que tiene que generar.
- Calcula los k vecinos más cercanos de los ejemplos de la clase minoritaria.
- Genera los ejemplos siguiendo este proceso:
    - Para cada ejemplo de la clase minoritaria, elige aleatoriamente el vecino a utilizar para crear el nuevo ejemplo.
    - Para cada atributo del ejemplo a sobre-muestrear, calcula la diferencia entre el vector de atributos muestra y el vecino elegido.
    - Multiplica esta diferencia por un número aleatorio entre 0 y 1.
    - Suma este último valor al valor original de la muestra.
    - Devuelve el conjunto de ejemplos sintéticos.

2. METODOLOGÍA

Para el desarrollo de este trabajo de investigación se utilizó como base el proceso de minería de datos; el método y la estrategia seguidos en el análisis comparativo se muestran en el diagrama de flujo en la figura 8.

Un conjunto de 745 tweets fueron recolectados para ser las variables independientes y la variable dependiente fue el resultado del análisis de las publicaciones que se hacían relacionadas con el Consejo Nacional Electoral del Ecuador y que se categorizaron en tres clases que fueron: positivas, neutrales y negativas. El siguiente paso fue transformar el conjunto de datos a formato ARFF que es el formato que lee el software Weka; posteriormente se procedió a obtener el Bag Of Words utilizando para ello el filtro denominado "StringToWordVector". Modelos basados en Naive Bayes, Logistic Regression, SMO, IBk, Decision Table, J48 y Random Forest fueron construidos para realizar la clasificación. Para abordar el desequilibrio de clase en los datos se utilzó la técnica sintética sobre muestreo de minoría (SMOTE). Los modelos fueron validados mediante la técnica de validación cruzada (10-fold–CV). Para comparar su desempeño respectivo,

se analizaron los resultados de las medidas de rendimiento que incluyeron exactitud, precisión, recall, el coeficiente de correlación de matthews MCC y el área ROC.

Una información más detallada sobre estos pasos se da en las siguientes subsecciones.

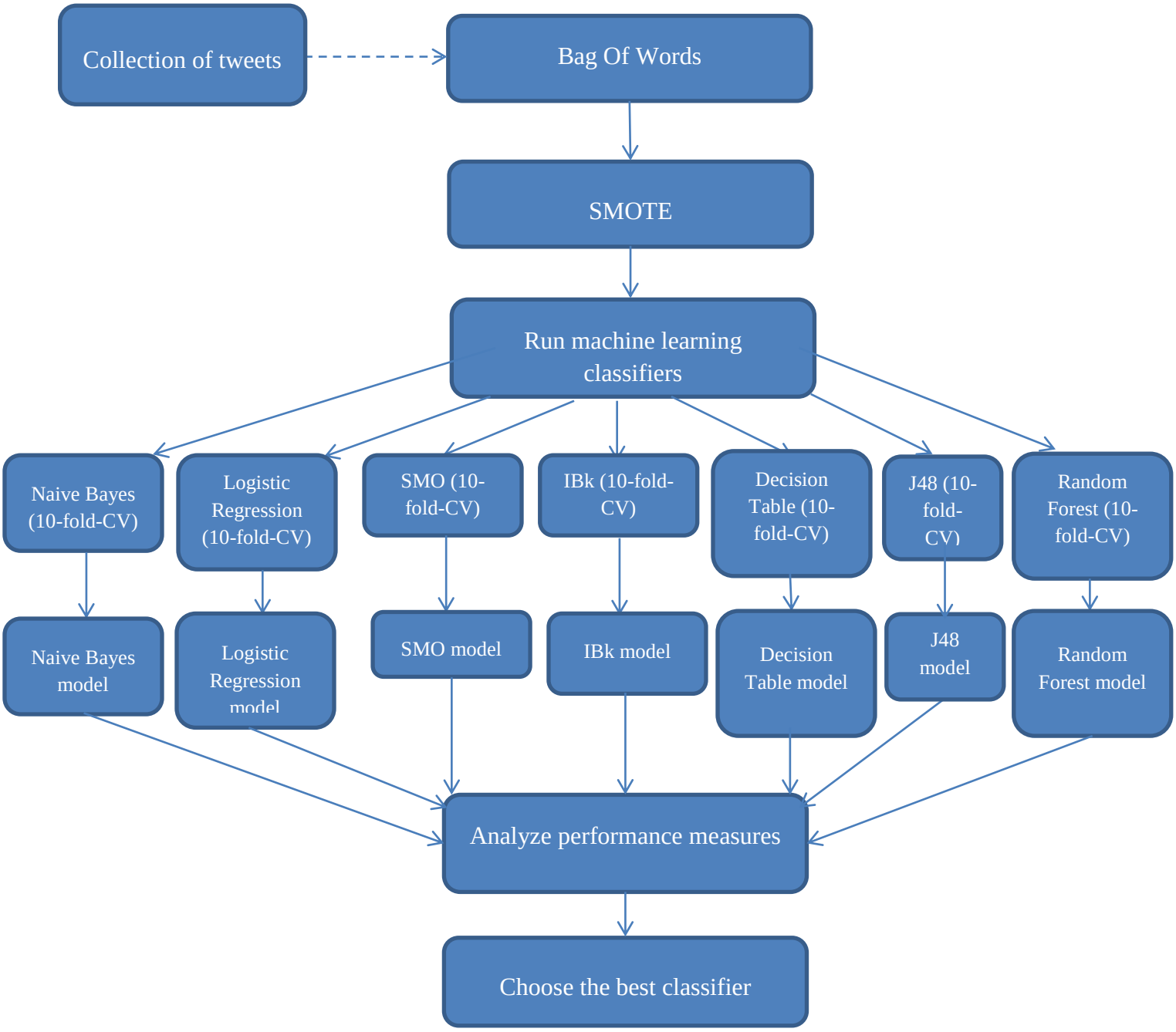

Figura 8. Configuración de la metodología

2.1. Elecciones Presidenciales 2017 en Ecuador

El pleno del Consejo Nacional Electoral cumpliendo con lo establecido dentro del cronograma electoral (Figura 9), el día 18 de octubre de 2017 convocó a los ecuatorianos a las elecciones generales 2017 conforme lo establecido en el artículo 1 de la convocatoria:

"Art1. A todas las ciudadanas y ciudadanos ecuatorianos mayores de dieciocho años con derecho a ejercer el voto, así como a aquellas personas mayores de dieciocho años de edad privadas de la libertad sin sentencia condenatoria ejecutoriada; y, de forma facultativa, a las ecuatorianas y ecuatorianos entre dieciséis y dieciocho años de edad, mayores de sesenta y cinco años, ecuatorianas y ecuatorianos domiciliados en el exterior debidamente registrados, integrantes de las Fuerzas Armadas y Policía Nacional en servicio activo, personas con discapacidad, extranjeras y extranjeros desde los dieciséis años de edad que hayan residido legalmente en el país al menos cinco años y se hubieren inscrito en el Registro Electoral, a elecciones, bajo las normas previstas en la Constitución de la República del Ecuador, Ley Orgánica Electoral y de Organizaciones Políticas de la República del Ecuador, Código de la Democracia, y Reglamentos expedidos por el Consejo Nacional Electoral, para elegir:

1. Presidenta o Presidente y Vicepresidenta o Vicepresidente de la República.

2. Cinco (5) representantes al Parlamento Andino.

3. Ciento treinta y siete (137) representantes a la Asamblea Nacional,…" (Consejo Nacional Electoral, 2016)

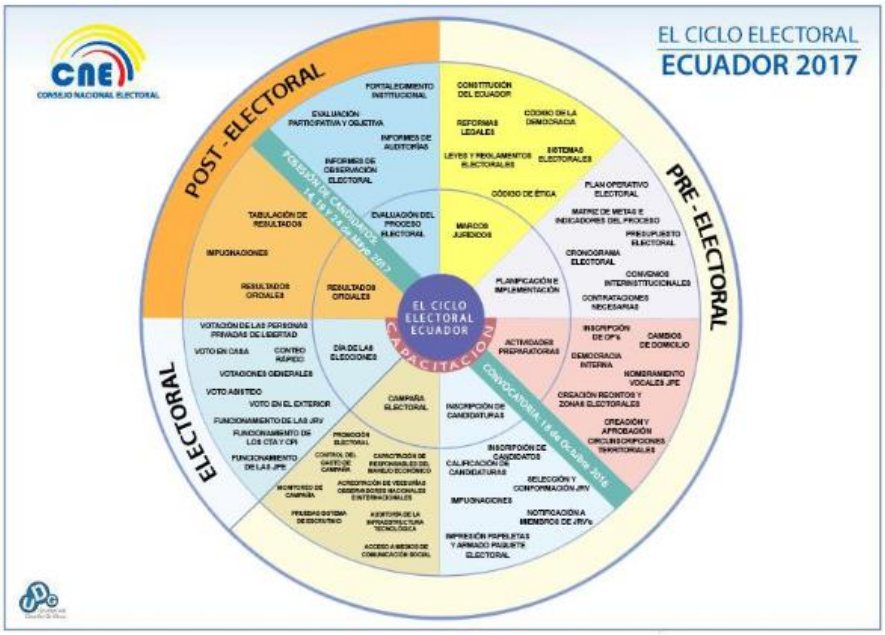

Figura 9. Cronograma Electoral Elecciones Generales 2017 (Consejo Nacional Electoral, 2016)

Para este proceso electoral fueron habilitados 12 816 698 electores, quienes acudieron a las urnas por primera vez el día 19 de febrero de 2017, elección cuyos resultados se los puede observar en la figura 10 y en la que el candidato con mayor votación no pudo obtener el porcentaje para ganar en una sola vuelta que es del 40% del total de votos válidos.

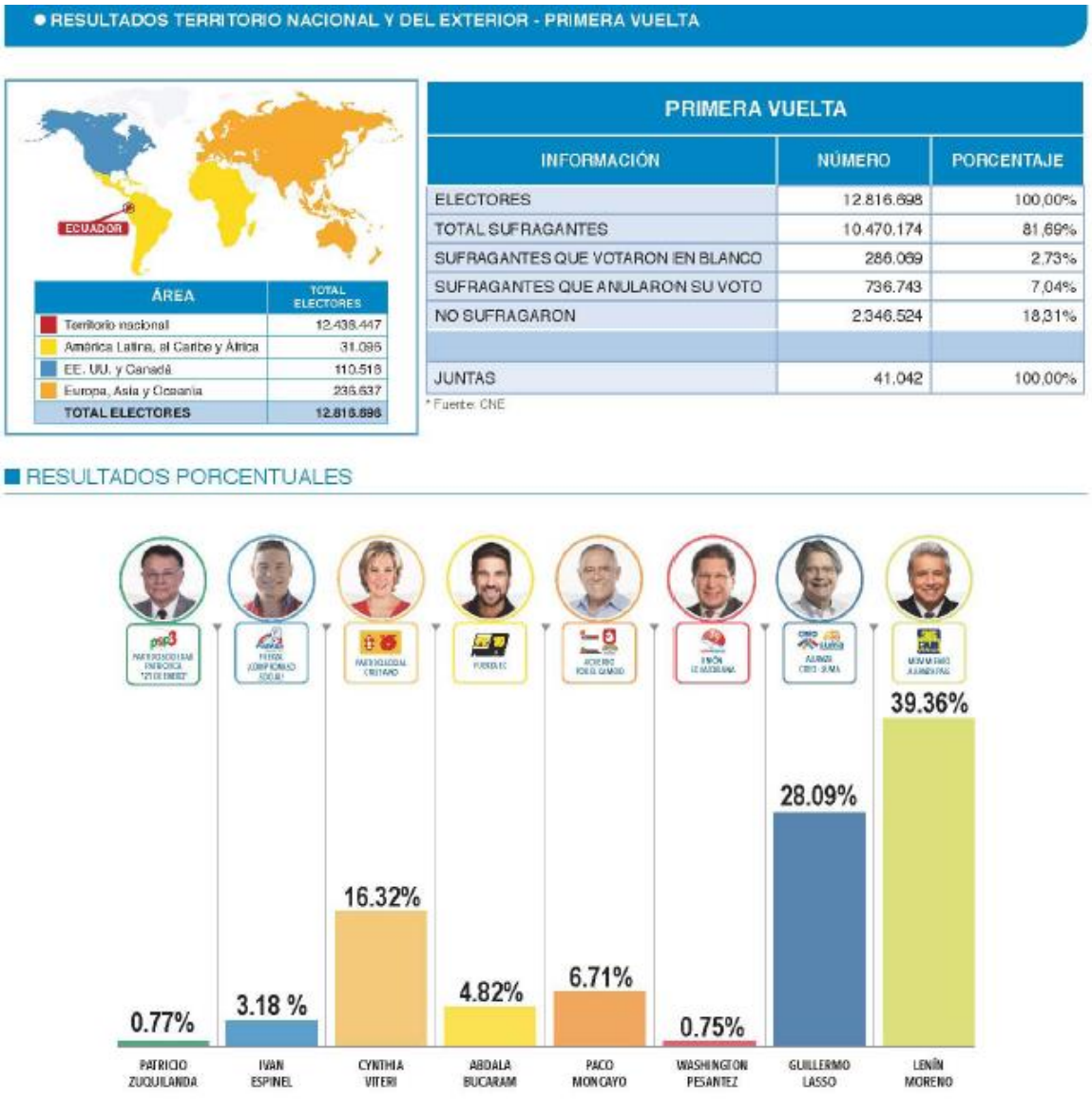

Figura 10. Resultados de candidatos a presidente primera vuelta (Consejo Nacional Electoral, 2017)

El día 2 de abril de 2017 se realizó el balotaje de la segunda vuelta entre los dos candidatos que obtuvieron la mayor votación en la primera vuelta, obteniendo los resultados descritos en la figura 11 dando como ganador al Lic. Lennin Moreno representante del partido Alianza PAIS, con un total de 5 062 018 votos equivalente al 51,16% de votos válidos.

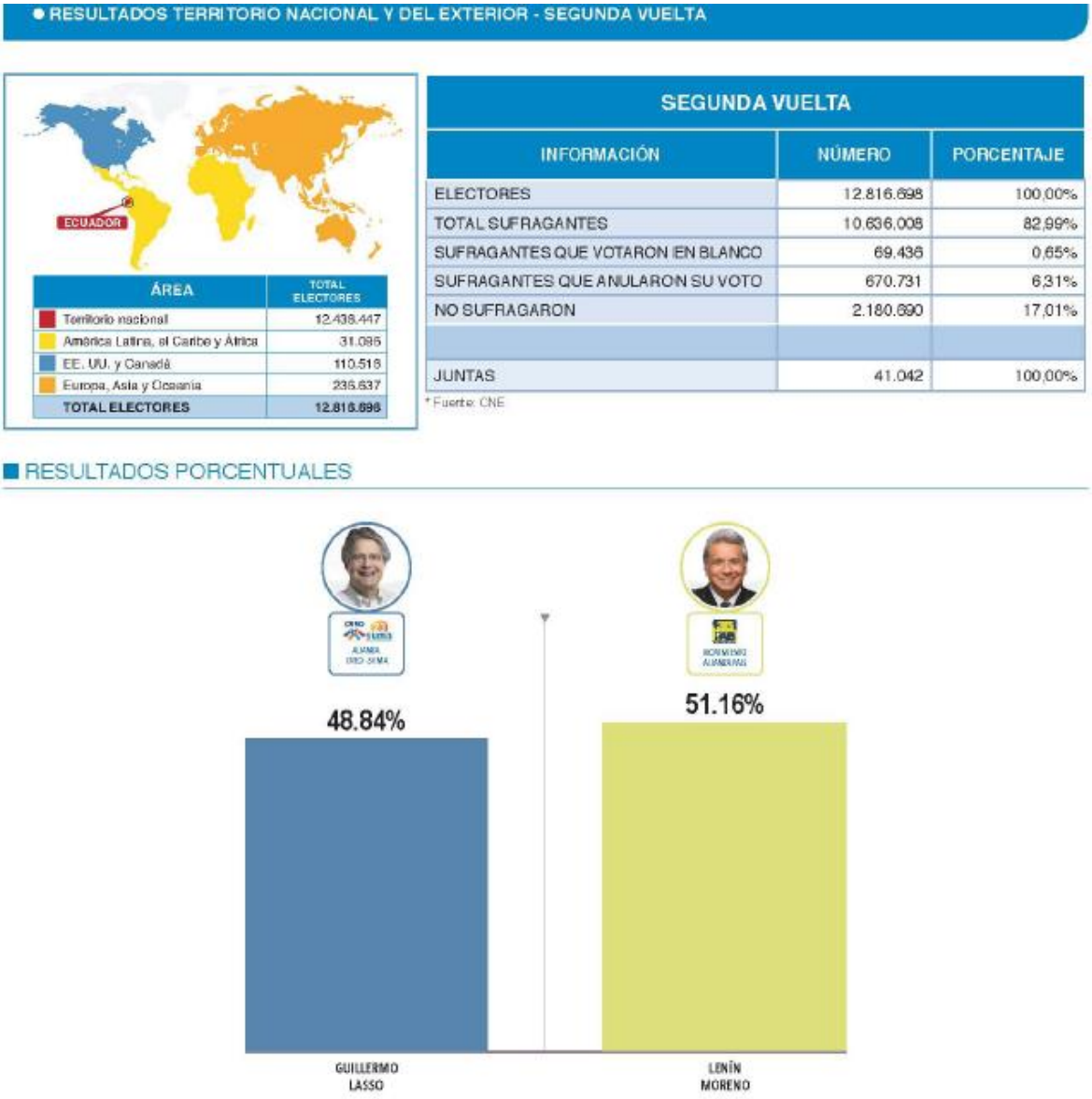

Figura 11. Resultados de candidatos a presidente segunda vuelta (Consejo Nacional Electoral, 2017)

2.2. Fase 1: Recolección de tweets

Los datos recolectados para realizar el presente trabajo, fueron tweets tomados de cuentas que durante el periodo pre-electoral y electoral de los comicios generales para elegir al nuevo Presidente del Ecuador hacían alusión al trabajo realizado por el Consejo Nacional Electoral.

La información en la actualidad es recolectada manualmente por el personal del área de comunicación de esta institución en la provincia de Loja y es ingresada en un archivo compartido en google drive con el formato que se puede observar en la figura12. En la columna tweet se copia el texto de los tweets realizados y que hacen alusión al CNE, en la columna cuenta se registra la cuanta desde la que se publicó el tweet, la columna elección hacer referencia a que si el tweet publicado tiene contenido relacionado con las elecciones presidenciales del año 2017 y finalmente la columna categoría registra la clasificación del tweet publicado asignándole una de las tres categorías que puede ser: positiva, negativa o neutal.

	TUIT	CUENTA	ELEECION	CATEGORIA
2	El candidato a la vicepresidencia por el movimiento CREO, ha cumplido actividades proselitistas desde el jueves,... fb.me/2AlyigGsD	@EcotelRadio	SI	NEUTRAL
3	Andres Páez en Loja pide a Correa declare bajo juramento que sus funcionarios no tienen dinero en paraísos fiscales	@EcotelRadio	SI	NEUTRAL
4	Hoy @CNELoja realiza pruebas técnicas del Sistema de Transmisión y Publicación de Resultados (STPR) para las #Elecciones2017EC.	@primereporte	SI	NEUTRAL
5	Jorge Bustamante: @PacoMoncayo supera las ideologías @CDLojaEc @RadioEkoos @primereporte @jimmyjairala	@EcotelRadio	SI	NEUTRAL
6	@PacoMoncayo: "en nuestra administración, ni el mote será pillo" @CDLojaEc @JennyMallaG @primereporte @RadioEkoos @lodelmomentoloj	@EcotelRadio	SI	NEUTRAL
7	Ahora | @cmontufarm , candidato a la Asamblea Nacional por Concertación/ visita #Loja, habla de sus propuestas.| @concertacionec	@CronicaLoja	SI	NEUTRAL
8	@CronicaLoja calificado por @CNELoja como proveedor para el próximo proceso electoral.	@CronicaLoja	SI	NEUTRAL
9	@MonseBustamant recordó sus orígenes en Macará @PacoMoncayo @eluniversocom @RadioEkoos @radiocityec @lodelmomentoloj @fmmundo	@EcotelRadio	SI	NEUTRAL
10	#Loja: Declaraciones de Jose Lucero, candidato a Asambleísta por Concertacion Loja. buff.ly/2h9gWBp	@RadioEkoos	SI	NEUTRAL
11	Entrevista a Sandra Jimenez Candidata a Asambleista por la Provincia de de Loja por el Partido Sociedad Patriotica fb.me/1DD0iXvgD	@CariamangaTv	SI	NEUTRAL
12	Entrevistando a Ramiro Armijos candidato a 2do. Asambleista por Sociedad Patriotica# 104.5 fm la radio/Loja	@camilaelizalde3	SI	NEUTRAL
13	#Loja | Candidato Asambleista Nacional cesarmontufar cumple Agenda de Planifacacion PreElectoral... instagram.com/p/BOAERxjj_Fg/	@alinstante2016	SI	NEUTRAL
14	Candidato Asambleista Nacional @cesarmontufar cumple Agenda de Planifacacion PreElectoral con Bases @51loja #Loja	@nixonperez77	SI	NEUTRAL

Figura 12. Formato xls de tweets recolectados

Una vez obtenido este archivo se procedió a realizar la depuración de los datos; primero se eliminaron columnas que se consideró innecesarias dejando solamente dos columnas que son: el tweet y la categoría.

Como segundo paso se eliminaron de la columna tweet todos los links o hipervínculos a páginas web que podía existir en cada tweet publicado, considerando que esta información no influye en la categorización del tweet y más bien podían empeorar el trabajo de los clasificadores.

Una vez depurado este archivo se procedió a realizar la transformación a formato ARFF, con dos atributos el tweet y la clase que podía ser positivos, negativos o neutros, el formato de este archivo se puede observar en la figura 13.

```
@relation Elecciones2017

@attribute tweet string
@attribute class {NEUTRAL,POSITIVA,NEGATIVA}

@data

"El candidato a la vicepresidencia por el movimiento CREO ha cumplido actividades proselitistas desde el jueves",NEUTRAL
"Andres Páez en Loja pide a Correa declare bajo juramento que sus funcionarios no tienen dinero en paraísos fiscales",NEUTRAL
"Hoy @CNELoja realiza pruebas técnicas del Sistema de Transmisión y Publicación de Resultados (STPR) para las #Elecciones2017EC.",NEUTRAL
"Jorge Bustamante: @PacoMoncayo supera las ideologías @CDLojaEc @RadioEkoos @primereporte @jimmyjairala",NEUTRAL
"@PacoMoncayo: en nuestra administración ni el mote será pillo @CDLojaEc @JennyMallaG @primereporte @RadioEkoos @lodelmomentoloj",NEUTRAL
"Ahora | @cmontufarm  candidato a la Asamblea Nacional por Concertación/ visita #Loja habla de sus propuestas.| @concertacionec",NEUTRAL
"@CronicaLoja calificado por @CNELoja como proveedor para el próximo proceso electoral.",NEUTRAL
```

Figura 13. Archivo arff

2.3. Fase 2: Obtención del Bag of Words

Una vez obtenido el archivo con formato ARFF, lo cargamos en el software weka utilizando la interfaz explorer y para obtener el bag-of-words aplicamos el filtro “StringToWordVector”; este filtro transforma atributos de cadena en vectores de palabras, es decir, crea un atributo para cada palabra que codifica la presencia o el recuento de palabras dentro de la cadena.

Se aplica este filtro para obtener palabras que se repitan algunas veces y que se relacionen directamente con las clases. El bag-of-words obtenido tiene un total de 3259 atributos y 745 instancias, tal como se lo observa en la figura 14.

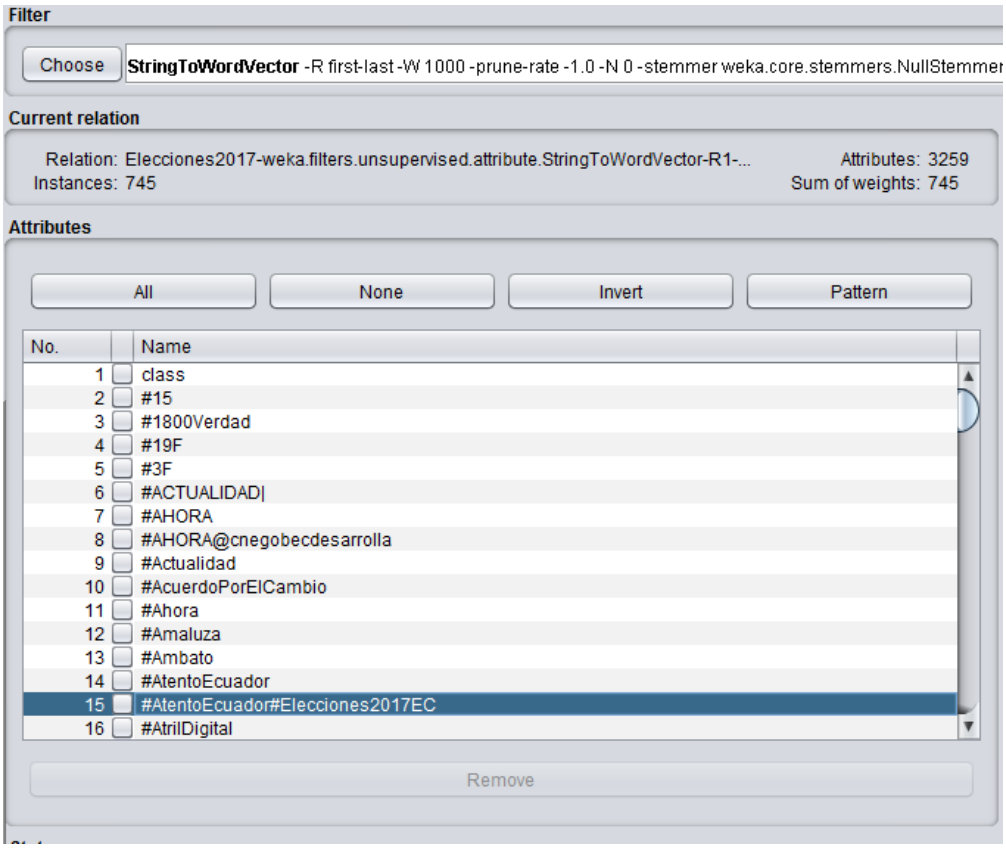

Figura 14. Bag of Words obtenido aplicando el filtro "StringToWordVector"

2.4. Fase 3: Equilibrar las clases con SMOTE

Un conjunto de datos está desequilibrado si las clases no están representadas más o menos igual. Esto es cierto en nuestros caso de estudio ya que 465 tweets fueron clasificados como neutros, 114 como positivos y 166 como negativos, figura 15.

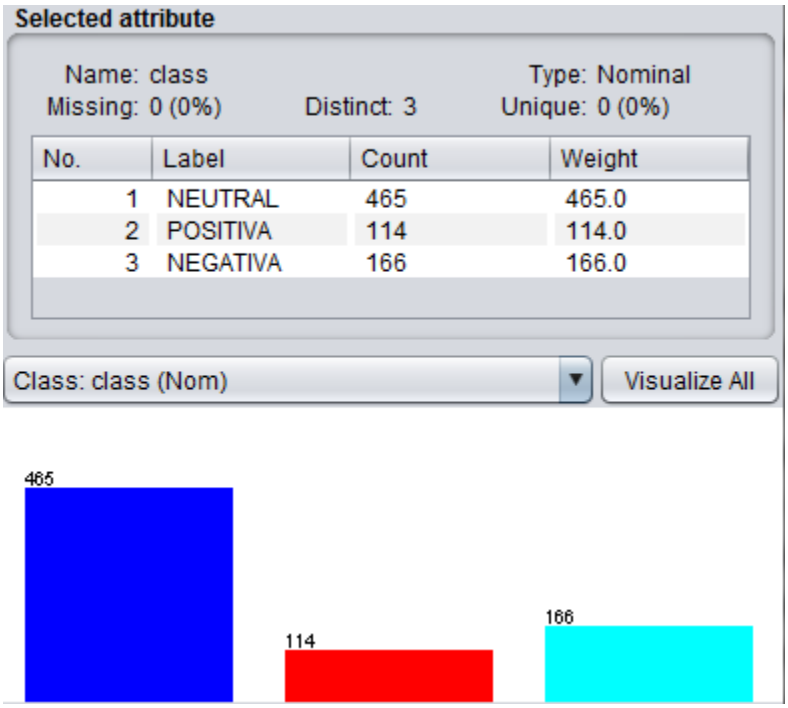

Figura 15. Clases base de datos original

Si el desequilibrio no se corrige puede dar lugar a niveles muy bajos en el recall y la precisión para medir el desempeño del clasificador. Para equilibrar el conjunto de datos por lo tanto aplicamos el enfoque SMOTE, una de las estrategias más utilizadas en la comunidad de aprendizaje de sistemas, para tratar con clases no balanceadas en problemas de clasificación.

Esta técnica sobre-muestra a la clase minoritaria creando ejemplos sintéticos en lugar de sobre-muestreo con reemplazo. Se administra tomando cada muestra de la clase minoritaria e introduciendo ejemplos sintéticos a lo largo de los segmentos lineales uniéndose a cualquiera de los k vecinos más cercanos de la clase minoritaria k más cercana. Dependiendo de la cantidad de muestras requeridas, vecinos del k más cercanos son seleccionados al azar.

SMOTE se ha utilizado con éxito para equilibrar las clases de problemas de clasificación con datos de una red social. Aquí, para nuestro caso al aplicar 3 iteraciones de SMOTE al conjunto de tweets los resultados fueron: 465 tweets para la clase neutral, 456 para la clase positiva y 332 para la clase negativa dando un total de 1253 instancias, figura 16.

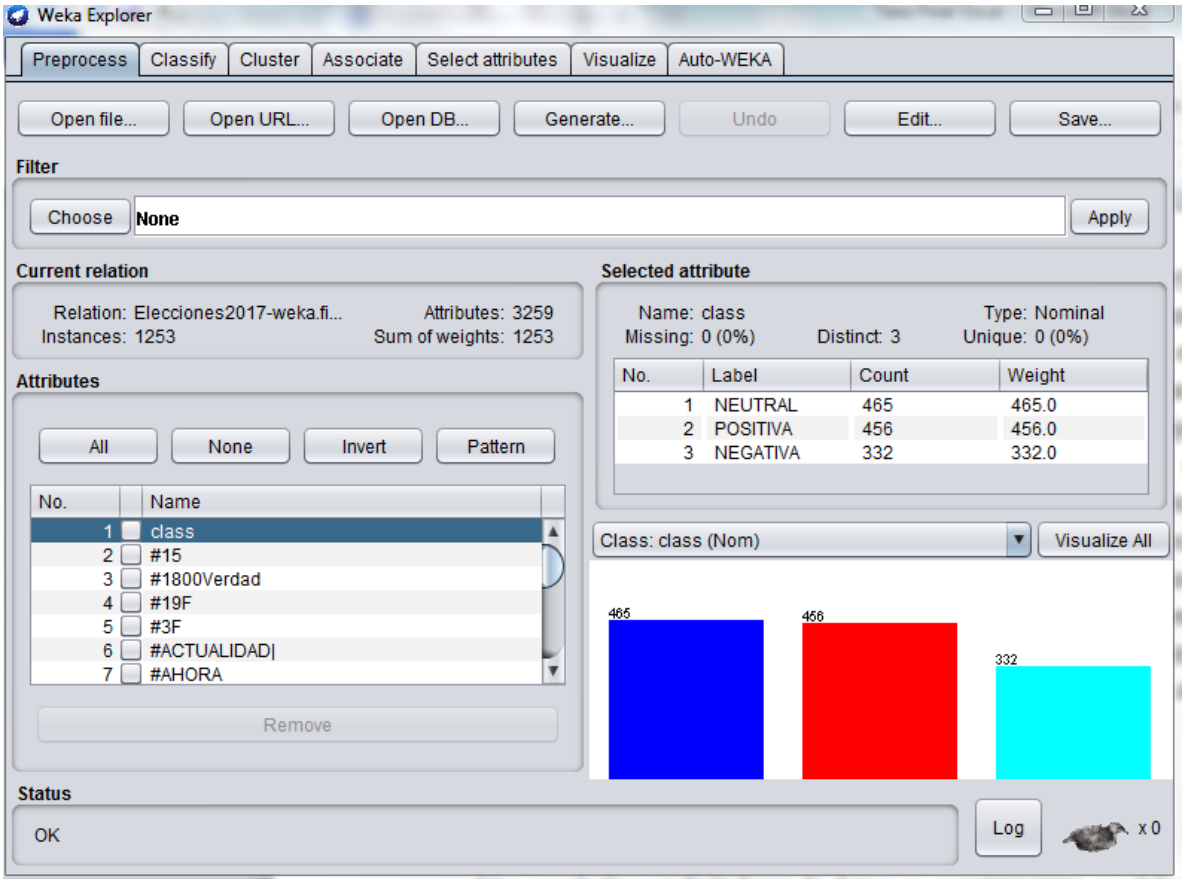

Figura 16. Datos una vez aplicada SMOTE

2.5. Fase 4: Ejecución de los clasificadores

Una vez balanceada la base de datos se construyeron modelos aplicando seis clasificadores Naive Bayes, Logistic Regression, SMO, IBk, Decision Table y Random Forest, con el objetivo de alcanzar el mayor porcentaje de acierto en la clasificación de los tweets en las tres clases definidas a priori.

Para evitar el desbordamiento, los modelos generados por los clasificadores en un principio nombrados se validaron utilizando la técnica de k-fold cross-validation (en nuestro estudio utilizamos 10-fold-CV). Esta técnica divide al azar la muestra original en 10 "pliegues" o submuestras. Una de las nueve submuestras es utilizada para probar el modelo, mientras que los nueve restantes se utilizan para el proceso de entrenamiento del algoritmo. Este proceso se repite 10 veces para cada una de las submuestras de k. Así, se obtienen los 10 resultados que luego se promedian para evaluar el desempeño del clasificador.

3. EVALUACIÓN Y RESULTADOS

3.1. Evaluación

Los clasificadores fueron evaluados usando las medidas estándar de exactitud, precisión, recall y el área bajo la curva ROC sugerido para pequeños conjuntos de datos figura 17. También se los evaluó con el coeficiente de correlación matthews (MCC), que a menudo se utiliza para medir el desempeño con bases de datos no balanceadas.

Performance Measure	Formula
Accuracy	$\frac{TP+TN}{TP+FP+FN+TN}$
Precision	$\frac{TP}{TP+FP}$
Recall	$\frac{TP}{TP+FN}$
MCC	$\frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP+FP)(TP+FN)(FP+TN)(FN+TN)}}$
ROC Area	$\frac{1}{2}\left(\frac{TP}{TP+FN}+\frac{TN}{TN+FP}\right)$

Figura 17. Medidas de desempeño para evaluar los clasificadores (Masías, y otros, 2016)

3.2. Resultados

En la tabla 1, se muestran los valores de las medidas de rendimiento para cada uno de los clasificadores aplicados sin utilizar el balanceo de clases SMOTE, donde se puede observar que el modelo con SMO es el que tiene los puntajes promedios más altos para la exactitud y recall, mientras que Logistic Regression tiene la más alta precisión y coeficiente MCC; Randon Forest obtiene la mejor Curva ROC.

Tabla 1. Medidas de rendimiento de los clasificadores sin SMOTE

Medida de rendimiento	Naive Bayes	Logistic	SMO	IBk	Decision Table	J48	Random Forest
Accuracy							
Neutral	0.671	0.748	**0.841**	0.634	0.946	0.858	0.981
Positiva	0.307	0.491	**0.254**	0.482	0.000	0.088	0.114
Negativa	0.735	0.590	**0.614**	0.084	0.373	0.416	0.217
(Avg.)	0.630	0.674	**0.701**	0.489	0.674	0.642	0.678
Precision							
Neutral	0.776	**0.819**	0.745	0.756	0.680	0.695	0.667
Positiva	0.255	**0.303**	0.326	0.165	0.000	0.192	0.591
Negativa	0.592	**0.726**	0.779	0.667	0.689	0.580	0.923
(Avg.)	0.655	**0.719**	0.688	0.646	0.578	0.592	0.712
Recall							
Neutral	0.671	0.748	**0.841**	0.634	0.946	0.858	0.981
Positiva	0.307	0.491	**0.254**	0.482	0.000	0.088	0.114
Negativa	0.735	0.590	**0.614**	0.084	0.373	0.416	0.217
(Avg.)	0.630	0.674	**0.701**	0.489	0.674	0.642	0.678
MCC							
Neutral	0.340	**0.463**	0.385	0.286	0.297	0.268	0.294
Positiva	0.135	**0.239**	0.177	0.029	-0.044	0.030	0.212
Negativa	0.549	**0.569**	0.617	0.182	0.415	0.374	0.396
(Avg.)	0.355	**0.452**	0.405	0.224	0.271	0.255	0.304
ROC Area							
Neutral	0.755	0.790	0.679	0.663	0.636	0.648	**0.800**
Positiva	0.656	0.677	0.621	0.530	0.472	0.526	**0.687**
Negativa	0.877	0.887	0.822	0.594	0.736	0.735	**0.906**
(Avg.)	0.767	0.794	0.702	0.627	0.633	0.649	**0.807**

En la figura 18, podemos observar el porcentaje de acierto global de cada uno de los clasificadores aplicados a la base de datos original es decir sin habar aplicado el balanceo con SMOTE, donde el las máquinas de soporte vectorial en weka SMO obtiene el mejor resultado con un 70,067%.

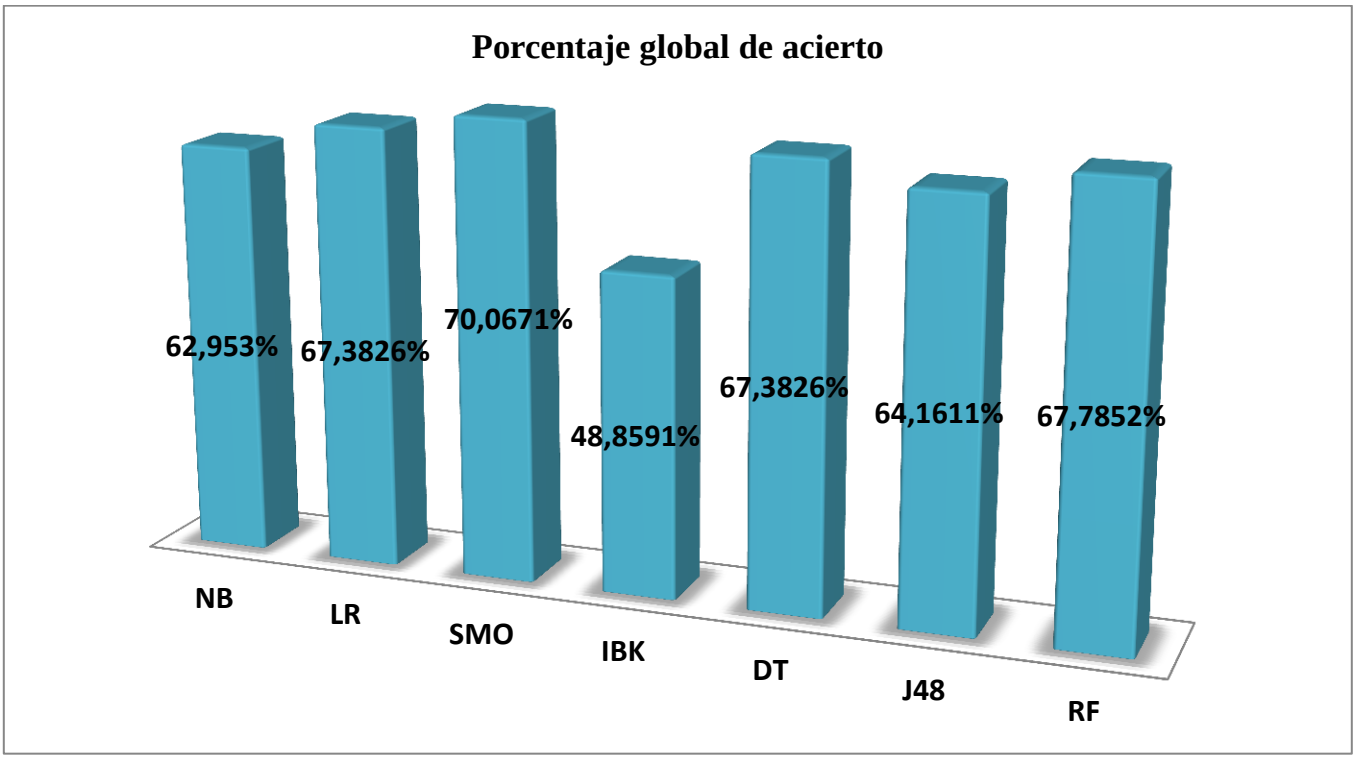

Figura 18. Porcentaje de acierto global de los clasificadores sin aplicar SMOTE

En la tabla 2, se muestran las valores de las medidas de rendimiento para cada uno de los clasificadores aplicados, tomando en cuenta que previamente se realizó el balanceo de clases con SMOTE, donde se puede observar que el modelo con SMO es el que tiene los puntajes promedios más altos para la accuracy, recall y MCC, mientras que Logistic Regression tiene la más alta precisión y Randon Forest obtiene la mejor Curva ROC.

Tabla 2. Medidas de rendimiento de los clasificadores usando SMOTE

Medida de rendimiento	Naive Bayes	Logistic	SMO	IBk	Decision Table	J48	Random Forest
Accuracy							
Neutral	0.776	0.794	**0.811**	0.254	0,927	0,811	0.972
Positiva	0.783	0.976	**0.980**	0.989	0,656	0,772	0.871
Negativa	0.889	0.979	**0.958**	0.946	0,307	0,729	0.768
(Avg.)	0.808	0.909	**0.911**	0.704	0.664	0,775	0.881
Precision							
Neutral	0.752	**0.963**	0.945	0.959	0,567	0,699	0.770
Positiva	0.913	**0.846**	0.873	0.564	0,857	0,850	0.973
Negativa	0.772	**0.945**	0.930	0.952	0,708	0,807	0.988
(Avg.)	0.816	**0.916**	0.915	0.813	0.710	0,783	0.902
Recall							
Neutral	0.776	0.794	**0.811**	0.254	0,927	0,811	0.972
Positiva	0.783	0.976	**0.980**	0.989	0,656	0,772	0.871
Negativa	0.889	0.979	**0.958**	0.946	0,307	0,729	0.768
(Avg.)	0.808	0.909	**0.911**	0.705	0.664	0,775	0.881
MCC							
Neutral	0.621	0.814	**0.812**	0.402	0,504	0,591	0.775
Positiva	0.769	0.852	**0.880**	0.552	0,636	0,710	0.880
Negativa	0.761	0.948	**0.923**	0.930	0,362	0,689	0.835
(Avg.)	0.712	0.863	**0.866**	0.596	0,514	0,660	0.829
ROC Area							
Neutral	0.883	0.967	0.888	0.625	0,825	0,846	**0.968**
Positiva	0.932	0.970	0.953	0.779	0,831	0,885	**0.985**
Negativa	0.939	0.996	0.977	0.966	0,744	0,885	**0.993**
(Avg.)	0.916	0.976	0.935	0.771	0,806	0,870	**0.981**

En cuanto al porcentaje global de aciertos figura 19, el modelo que tuvo mayor porcentaje de acierto fue la máquina de soporte vectorial (SMO), seguido muy de cerca por el algoritmo Logistic Regression quien también tuvo un buen porcentaje de aciertos, mientras que las tablas de decisión (Decision Table) fue el que obtuvo menor porcentaje de acierto.

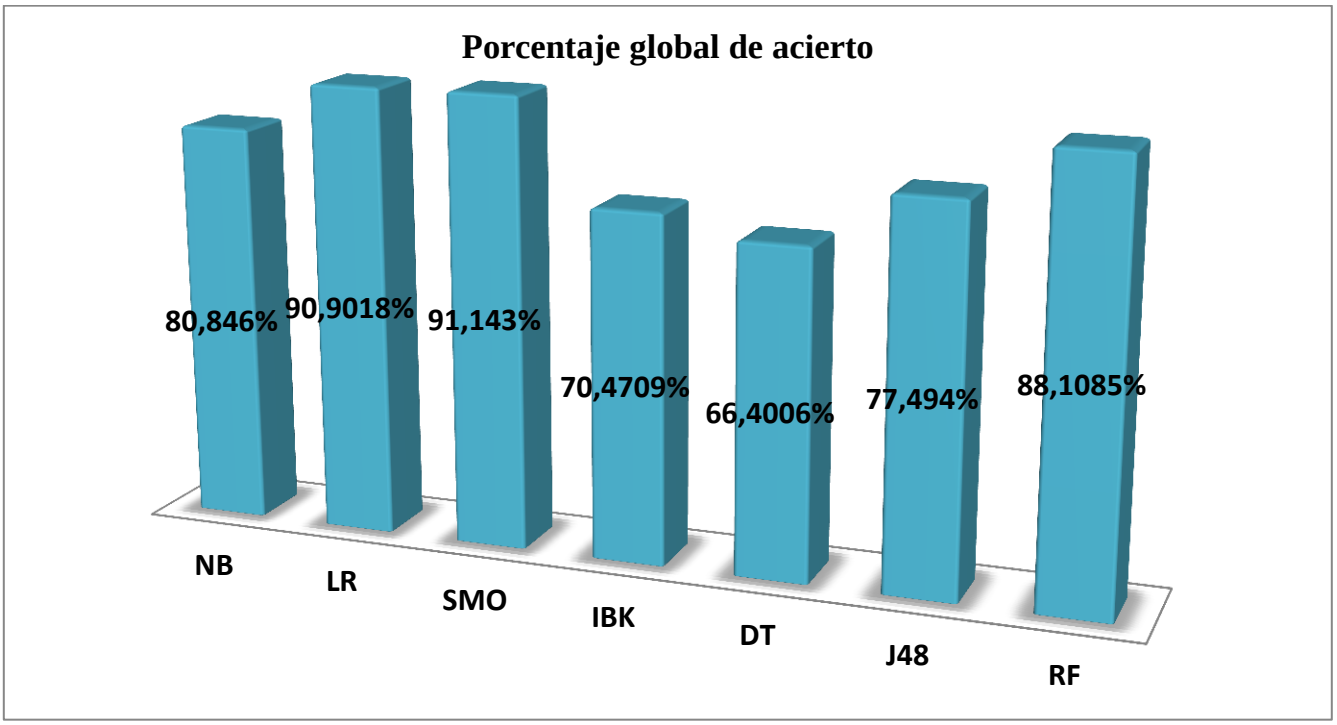

Figura 19. Porcentaje global de acierto de algoritmos de clasificación aplicando SMOTE

3.3. Análisis de resultados

Sin realizar ninguna iteración de SMOTE, los clasificadores entregan peores resultados (Figura 18). Vamos a analizar estos resultados tomando la medida de rendimiento recall o en español sensibilidad (se refiere a la fracción de ejemplos de la clase de todo el conjunto que se clasifican correctamente), como base para este análisis.

Tabla 3. Recall en base de datos sin realizar ninguna iteración de SMOTE

Medida de rendimiento	Naive Bayes	Logistic	SMO	IBk	Decision Table	J48	Random Forest
Recall							
Clase Neutral	0.671	**0.748**	**0.841**	**0.634**	**0.946**	**0.858**	**0.981**
Clase Positiva	0.307	0.491	0.254	0.482	0.000	0.088	0.114
Clase Negativa	**0.735**	0.590	0.614	0.084	0.373	0.416	0.217

Al tener 465 tweets clasificados como neutros, 114 como positivos y 166 como negativos (Figura 15), los clasificadores tienen un fácil sesgo hacia la clase mayoritaria, es decir la tasa de error del clasificador no es representativa de lo bien que realiza la tarea.

Por ejemplo si observamos el Recall del clasificador SMO en la Tabla 3, el algoritmo clasifica el 84,1% de muestras como clase Neutral, 25,4% de muestras como clase Positiva y 61,4% como clase negativa; a pesar de que se tenga un porcentaje alto de clasificación de la clase neutral, esto no significa que sea un buen clasificador, pues contrariamente tuvo un 75,6% de error en la clasificación de las muestras de la clase positiva.

¿Pero qué pasa al aplicar 3 iteraciones de SMOTE a la base de datos y obtener 465 muestras para la clase neutral, 456 para la clase positiva y 332 para la clase negativa dando un total de 1253 instancias (Figura 16) y aplicar los clasificadores?

Sucede que los porcentajes de acierto aumentan con respecto de la aplicación de estos mismos algoritmos a la base de datos original (Figura 15), ahora por ejemplo en este caso Naive Bayes alcanza un 80,84%, SMO obtiene un 91,14%.

Tabla 4. Recall en base de datos aplicando SMOTE

Medida de rendimiento	Naive Bayes	Logistic	SMO	IBk	Decision Table	J48	Random Forest
Recall							
Clase Neutral	0.776	0.794	**0.811**	0.254	0,927	0.811	0.972
Clase Positiva	0.783	0.976	**0.980**	0.989	0,656	0.772	0.871
Clase Negativa	0.889	0.979	**0.958**	0.946	0,307	0.729	0.768

Como se puede observar en la Tabla 4 al balancear las clases con la aplicación de SMOTE, el recall para las clases positiva y negativa mejora a costa de asumir un peor recall de la neutral; por ejemplo si observamos el recall del clasificador SMO (Tabla 4), el clasificador clasifica el 81,1% de muestras como clase neutral, 98% de las muestras como de la clase positiva y 95,8% de las muestras como clase negativa; es decir la tasa de error del clasificador es representativa de lo bien que realiza su tarea.

Con la interfaz KnowledgeFlow de weka (Figura 20) se obtuvo las curvas ROC de la aplicación del clasificador SMO tanto en la base de datos sin aplicar SMOTE (Figura 21), como en la aplicación del algoritmo a la base de datos aplicando tres iteraciones de SMOTE (Figura 22).Los gráficos de las curcas ROC fueron realizados tanto para la clase neutral, positiva y negativa en cada caso.

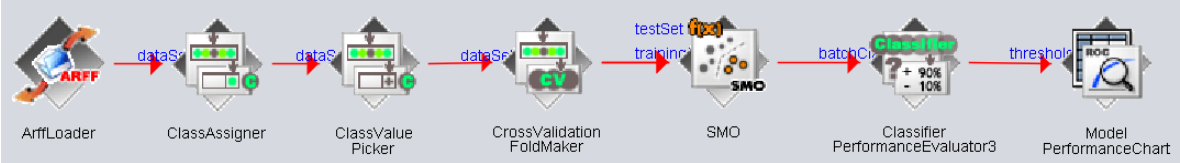

Figura 20. Configuración de la interfaz KnowledgeFlow en weka para obtener curvas ROC

Como se puede observar en la figura 21 la clase Negativa tiene una mejor curva ROC que la clase Positiva y Neutral, teniendo una diferencia significativa del área bajo su curva, estos resultados son los obtenidos al aplicar el algoritmo SMO en la base de datos desbalanceada es decir sin haber aplicado SMOTE.

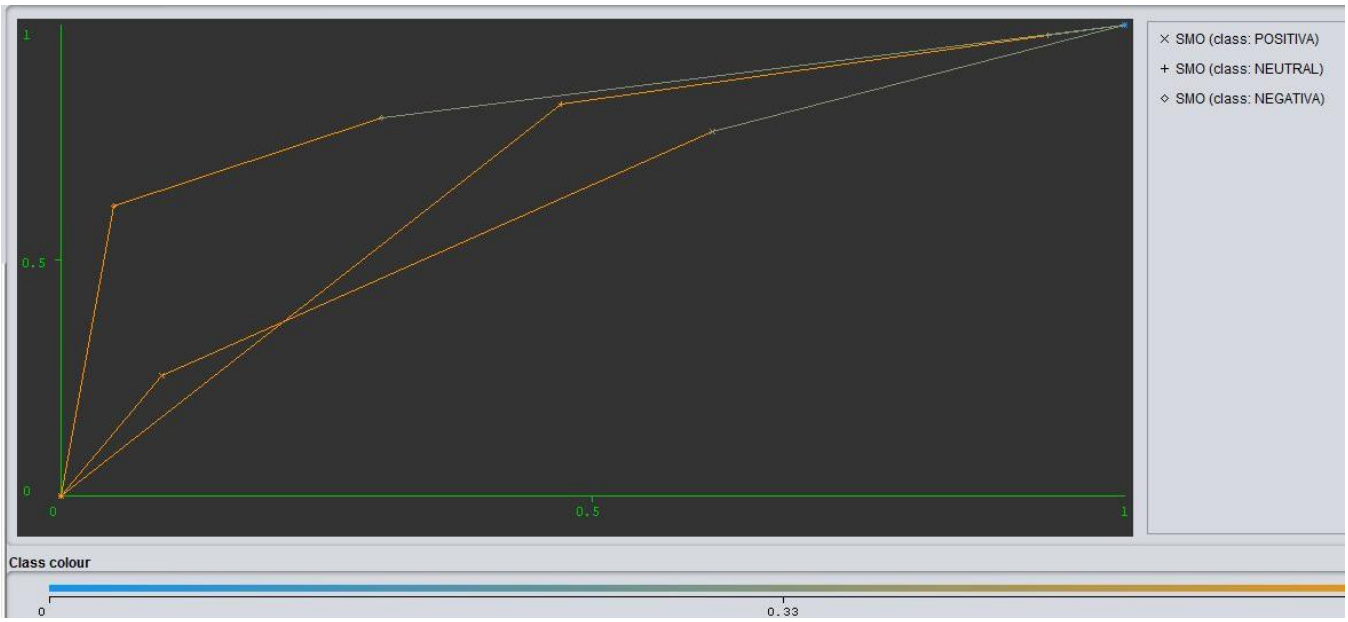

Figura 21. Curva ROC de la aplicación de SMO para las clases neutral, positiva y negativa sin aplicar SMOTE

La figura 22 muestra las curvas ROC al haber aplicado las máquinas de soporte vectorial en weka SMO a la base de datos balanceada con SMOTE, aquí el área bajo las curvas de las clases Positiva y Negativa son similares y la diferencia con la curva ROC de la clase Neutral no es significativa.

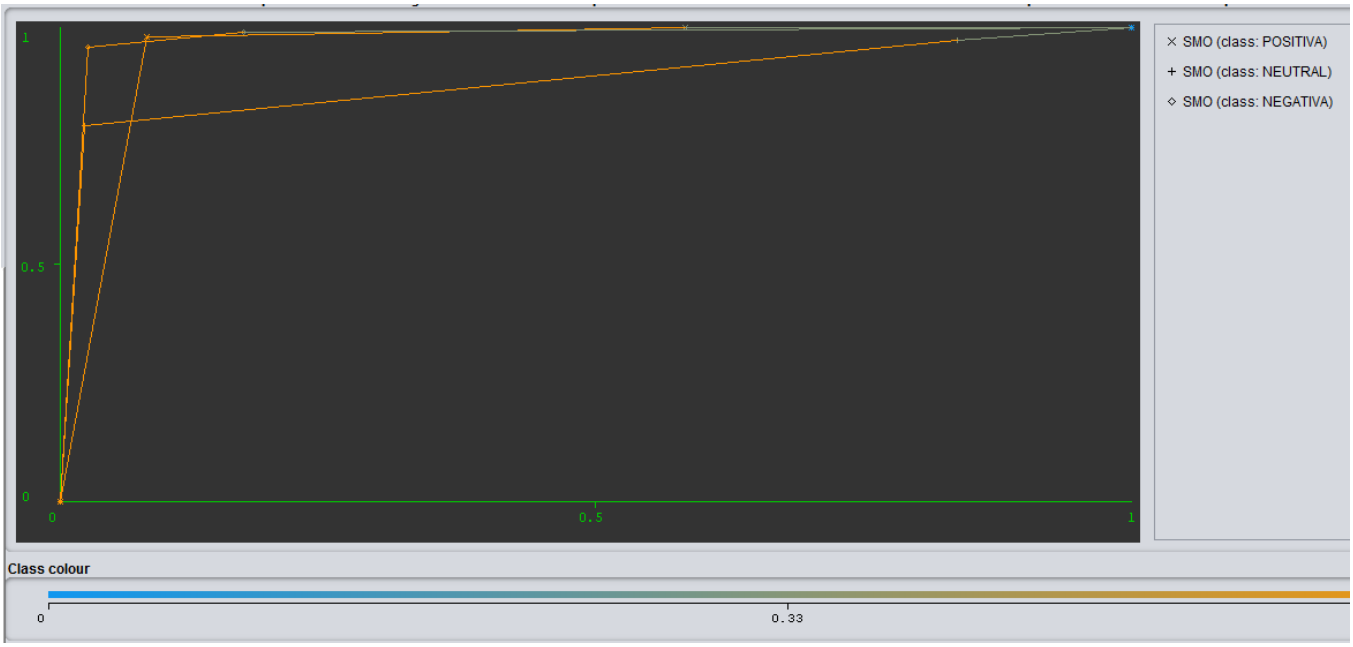

Figura 22. Curva ROC de la aplicación de SMO para las clases neutral, positiva y negativa aplicando SMOTE

Si comparamos la Figura 21 y Figura 22, podemos observar que las curvas ROC mejoran en la Figura 22 con respecto de la Figura 21, ya que estas son las curvas de las clases neutral, positiva y negativa resultado de la aplicación de los algoritmos a la base de datos que fue balanceada con SMOTE; así mismo podemos observar que las curvas en la Figura 22 tienen punto de corte cercanos a 1, con lo que se comprueba que al aplicar SMOTE a la base de datos no solamente se mejora el porcentaje global de acierto sino también se mejora cada una de las medidas de rendimiento para cada una de las clases y no se sesga estas medidas para la clase mayoritaria.

4. TRABAJOS FUTUROS

De la bibliografía revisada se observó que existen varios estudios de investigación que engloban la normalización de tweets, por lo que el problema planteado anteriormente podría mejorarse añadiendo algún proceso de normalización antes de tratar los tweets.

Otra mejora que puede añadirse al modelo desarrollado, es la recolección automática de los tweets, utilizando herramientas pagadas o API´s para esta recolección y aumentar así el número de tweets de la base de datos.

Un trabajo futuro que puede realizarse es el de añadir nuevas categorías a predecir, como por ejemplo: Informativa, Muy Positivo, Muy Negativo. En este caso, habría que tener en cuenta las

estrategias de intensificación y atenuación existentes en gramática. Quizá añadir como atributos el número de retweet de un mensaje o el número de hashtags que contiene puede ayudar al clasificador.

5. CONCLUSIONES

- Se ha identificado que las Máquinas de Soporte Vectorial y la Regresión Logística, han demostrado ser los modelos más idóneos para este problema de clasificación, pudiéndose implementar en la categorización de tweets de tendencia política.
- Se puede excluir a las Tablas de Decisión para la categorización de tweets, considerando que fueron los que menor porcentaje de acierto global obtuvieron en el presente estudio.
- Los resultados muestran que los modelos generados por la Regresión Lineal presentan niveles bajos de error en los resultados de la clasificación, una señal de que SMOTE combinado con LR ayuda a incrementar la capacidad predictiva de los modelos de las clases.

6. BIBLIOGRAFÍA

Política en Tweets. (19 de Febrero de 2017). Recuperado el 2017 de Junio de 2017, de https://twitter.com/politicaentweet/status/829506712004620288

Amaya de la Peña, I. (2015). *Presencia en Twitter de los candidatos a laas elecciones madrileñas de 2015.* Madrid: Universidad Politécnica de Madrid.

Batista, F., & Ribeiro, R. (2012). The L2F strategy for sentiment analysis and topic classification. *TASS 2012 working notes*.

Blanco, E. J., & Hermida, S. (2016). *Algoritmos de clustering y aprendizaje automático aplicados a Twitter.* Universidad Politécnica de cataluña.

Buill Vilches, J. (19 de Junio de 2014). Clasificación automática de textos y explotación BI. Barcelona, España.

Calvo Vilares, D. (2014). *Análisis de contenidos en Twitter: clasificación de mensajes e identificación de la tendencia pol´ıtica de los usuarios.* Universidad de la Coruña.

Chang, C.-C., & Lin, C.-J. (2011). LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 27.

Consejo Nacional Electoral. (2016). *Certificación ISO/TS 17582:2014 ISO 9001:2008 Objetivo y Desafío de la Institucionalidad Electoral.* Quito: Dirección Nacional de Comunicación Electoral.

Consejo Nacional Electoral. (18 de Octubre de 2016). *Covocatoria a Elecciones 2017.* Recuperado el 15 de Junio de 2017, de http://cne.gob.ec/images/d/2016/Elecciones_2017/CONVOCATORIA_ELECCIONES_2017.pdf

Consejo Nacional Electoral. (13 de Febrero de 2016). *Plan Operativo Elecciones Generales 2017.* (C. N. Electorales, Ed.) Recuperado el 15 de Junio de 2017, de http://cne.gob.ec/images/d/2016/Elecciones_2017/PlanOperativoEleccionesGenerales2017.pdf

Consejo Nacional Electoral. (2017). *Resultados Electorales 2017.* Quito.

Consejo Nacional Electoral del Ecuador. (19 de Febrero de 2017). *Consejo Nacional Electoral*. Recuperado el 15 de Junio de 2017, de https://resultados2017.cne.gob.ec

Echegoyen, C. (2007). Clasificación de Textos. *echegoyenclasificacion*, 3-4.

Gálvez Pérez, J. R. (2015). Sistema automático para la clasificación de la opinión pública generada en Twitter. *Research in Computing Science*, 23-36.

Godino Martínez, A. (2014). *Sistema de clasificación automática sobre streams de tweets.* Universidad Carlos III de Madrid.

González Rubio, C. (2015). *Clasificación Automática de Texto para el Seguimiento de Campañas Electorales en Redes Sociales.*

Hosmer Jr, D. W. (2013). *Applied logistic regression.* John Wiley \& Sons.

Jiménez, M. G., & Álvarez Sierra, A. (2010). Análisis de datos en WEKA–pruebas de selectividad. *línea] disponible en http://www. it. uc3m. es/jvillena/irc/practicas/06-07/28. pdf*.

Joanneum, F. (2005-2006). Cross-Validation Explained.

Lage García, L. (2014). Herramienta para el análisis de la opinión en tweets periodísticos.

López, J. M., & Herrero, J. G. (2006). Técnicas de análisis de datos. *Aplicaciones Prácticas utilizando Microsoft Excel y WEKA*, 2016.

Martínez Cámara, E., García Cumbreras, M. Á., Martín Valdivia, M. T., & Ureña López, L. A. (2013). Sinai en tass 2012. *Procesamiento del Lenguaje Natural*, 53-60.

Martínez, G. (2001). Minería de datos. *Cómo hallar una aguja en un pajar*, 18.

Martis Cáceres, M. A. (Enero de 2012). Clasificación Automática de la Intención del Usuario en Mensajes de Twitter.

Masías, V., Valle, H., Morselli, M., Crespo, C., Vargas, F., Laengle, A., y otros. (2016). Modeling Verdict Outcomes Using Social Network Measures: The Watergate and Caviar Network Cases. *PloS one*, 10.

Moreno, J., Rodríguez, D., Sicilia, M., Riquelme, J., & Ruiz, R. (2009). SMOTE-I: mejora del algoritmo SMOTE para balanceo de clases minoritarias. *Actas de los Talleres de las Jornadas de Ingeniería del Software y Bases de Datos*, 75.

Moya Sánchez, M., & Herrera Damas, S. (2015). CÓMO PUEDE CONTRIBUIR TWITTER A UNA COMUNICACIÓN POLÍTICA MÁS AVANZADA. *Arbor*, 257.

Neyra, L. (2016). *Categorización Automática De Respuestas Aplicando Algoritmos De Clasificación Supervisada Al Análisis De Las Contestaciones De Estudiantes A Una Serie De Preguntas Tipo Test.* España: Universidad del País Vasco.

Pla, F., & Hurtado, L.-F. (2013). ELiRF-UPV en TASS-2013: Análisis de sentimientos en Twitter. En *XXIX Congreso de la Sociedad Espanola para el Procesamiento del Lenguaje Natural (SEPLN 2013). TASS* (págs. 220-227).

Platt, J. C. (1999). 12 fast training of support vector machines using sequential minimal optimization. *Advances in kernel methods*, 185-208.

Ramírez de la Rosa, A. G. (Noviembre de 2010). Clasificacion de textos utilizando información inherente al conjunto a clasificar. Tonantzintla, Puebla, México.

Soltero Domingo, F. J., & Bodas Sagi, D. J. (2012). Clasificadores inductivos para el posicionamiento web. *El profesional de la información*, 4-13.

Taborda, C. H., García Gelvez, N., & Rozo Palacios, J. J. (2016). Análisis de datos mediante el algoritmo de clasificación J48, sobre un cluster en la nube de AWS. *Redes de Ingeniería*, 3-15.

Tomás Diaz, D. (2010). Sistemas de clasificación de preguntas basados en corpus para la búsqueda de respuestas.