PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA INFORMÁTICA

FORMAS DE REPRESENTACIÓN DE TEXTOS CORTOS PARA SU POSTERIOR CLASIFICACION AUTOMATICA

RAÙL IGNACIO SANHUEZA ARANCIBIA

Profesor Guía: **Rodrigo Alfaro Arancibia**

TRABAJO DE TESIS DE GRADO
PARA OPTAR AL TÍTULO PROFESIONAL DE
INGENIERO CIVIL EN INFORMÁTICA

Diciembre, 2013

Índice

Resumen

La presente tesis tiene como finalidad describir formas de representar textos cortos para su posterior clasificación automática.

Se presentan la introducción de la clasificación automática de textos y luego se describen las formas representar textos, para de esta manera poder a través de algoritmo de aprendizaje automático realizar una clasificación automática.

En el desarrollo se describe la herramienta a utilizar para realizar las pruebas, además de la descripción de los datos, estos generan los gráficos que aportan una mayor facilidad de análisis de los resultados, comparándolo con las métricas descritas ocupadas con los métodos de representación abordada en esta investigación.

Se concluye la investigación con el análisis de los resultados y con la descripción de la mejor descripción de los mejor o mejores métodos de representación de textos cortos, además con los trabajos propuestos a futuro.

Palabas-claves: clasificación automática de texto.

Lista de Figuras

Lista de Tablas

1. Introducción

La inteligencia artificial es hoy alcanzable para todas las personas, ya que muchas aplicaciones y programas la ocupan, lo que antiguamente se veía solo en ficción es hoy un campo muy utilizado en la informática actual. Una de las ramas de la inteligencia artificial es el aprendizaje automático, el cual se inspira en el humano imitando su manera de aprender, para ser aplicado en las maquinas. Esto se puede apreciar en variados sitios web, como los más populares facebook, youtube, etc., los cuales aprenden de un perfil de usuario, sus preferencias mostrando, en el caso de youtube, los videos relacionados a las búsquedas antiguamente ha realizado, de esta manera va sugiriendo, según las preferencias del usuario, lo videos que tengan alta probabilidad de ser vistos por el usuario.

El modelado del lenguaje natural, es la manera en que la maquina logra distinguir e identificar la forma en que nos comunicamos, de manera de lograr clasificar de automáticamente el mensaje entregado. Es un desafío para la inteligencia artificial lograr que la maquina "entienda" lo que lee o escucha, ya que hay métodos que logran analizar el lenguaje, pero aún están lejos de lograr el objetivo, es que un software entienda.

Según la forma de representación de los mensajes o datos entregados, implica directamente cual va a ser la complejidad de su análisis por medio de algoritmos, es decir mientras más sencilla la forma de representar un texto, más complejo va a ser el algoritmo que tenga la función de clasificar dicho texto, agregando costos de tiempo, memoria, etc.

Por lo dicho anteriormente que se realiza la esta investigación, con el propósito de buscar métodos de representación de textos costos, ya que actualmente, las redes sociales tiene gran influencia en el mundo moderno, entregando gran cantidad de datos en forma de lenguaje natural, es por eso que se ve la necesidad de poder resumir esta información, de manera de lograr un análisis más certero de las preferencias o emociones de las personas, ya que tiene la ventaja de que sus publicaciones son al momento que el usuario tienen la emoción y la necesidad de publicarlo, ya que al contrario con una encuesta, la emoción puede pasar hasta llegar a ser otra totalmente distinta. La red social que se analiza en la memoria es Twitter, la cual tiene más de 200 millones de usuarios, generando 65 millones de tweets diarios, en un límite de 140 caracteres, la representación de estos tweets necesita ser mejorada para que los algoritmos puedan realizar un análisis más complejos, mas optimizarlos en tiempo y costos, de manera de aumentar su probabilidad de acierto, y lograr, como objetivo final, poder realizar análisis certero de los usuarios de esta red social.

1.1 Objetivos

A continuación se describe qué se pretende de la investigación, expresado en el objetivo general y específico.

1.1.1 Objetivo General

Analizar las formas de representación de textos cortos para su posterior clasificación automática, para proponer la mejor alternativa para clasificación de estos usando un enfoque de máquinas de aprendizaje y representación vectorial.

1.1.2 Objetivo Especifico

- Explorar y comprender sobre la clasificación automática de textos.
- Revisar métodos para comparar la capacidad de clasificación y aprendizaje.
- Describir e indagar sobre las formas de representación de textos.
- Diferenciar entre las formas de representar textos para pronosticar a gran rasgo las mejores para su análisis.
- Formular un plan de pruebas y análisis para los clasificadores junto con las formas de representación.
- Proponer la mejor forma de representar textos cortos en vectores, cuantificando con resultados obtenido en las pruebas.

2. Marco Teórico.

En este capítulo se presentan los conceptos utilizados durante esta investigación de tesis. Se introduce a la clasificación automática de textos y además se presentan estudios relacionados que conforman la base de inspiración y comparación.

2.1 Clasificación Automática de Textos

La clasificación automática de textos últimamente ha recibido más atención debido al incremento en la cantidad de información disponible en formato electrónico. Es por esta razón que cada vez es mayor la información que busca, y además encontrar ésta en un tiempo adecuado [1].

El objetivo de la clasificación automática de texto es categorizar documentos dentro de un número fijo de categorías predefinidas en función de su contenido. Un mismo documento puede pertenecer a una, varias, todas o ninguna de las categorías dadas [2]. Cuando se utiliza aprendizaje, el objetivo es aprender a clasificar a partir de ejemplos que permitan hacer la asignación a la categoría automáticamente.

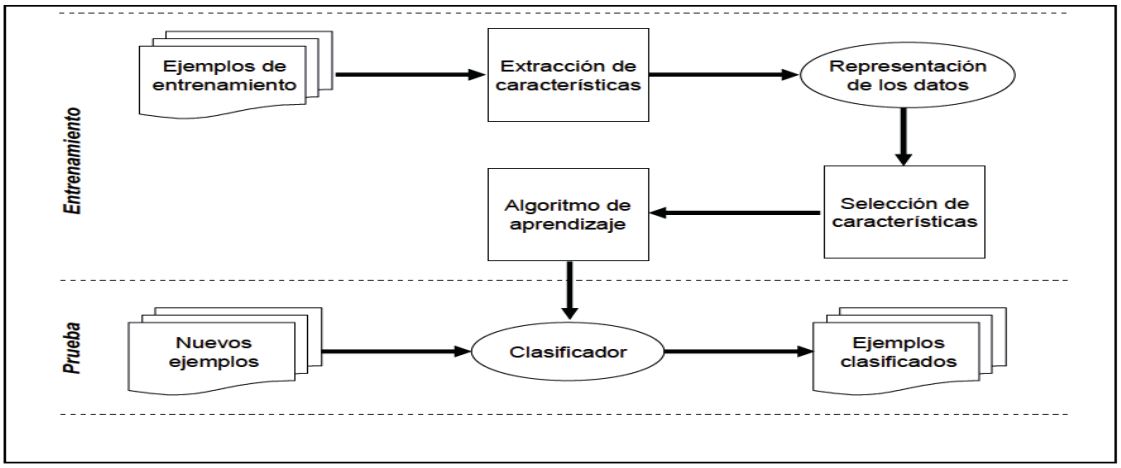

Figura 2-1 Paradigma de aprendizaje

Definir la clasificación de textos de manera formal Sebastiani [3] plantea que la tarea es aproximar la función objetivo desconocida, que describe como un documento debe ser clasificado, por medio de una función llamado clasificador de manera que ambas funciones coincidan lo más posible.

Las categorías corresponden solo a etiquetas simbólicas, no existe conocimiento adicional de sus significados para ayudar a construir el clasificador. Además sólo se utiliza el texto del documento para poder clasificar estos en las categorías, no haciendo uso de algún tipo de conocimiento exógeno, como por ejemplo metadatos con información como la fecha de publicación, tipo de documento, fuente de publicación, entre otros.

La construcción de un clasificador automático de texto comienza con la recopilación y clasificación manual de un conjunto de documentos (documentos de entrenamiento), después se llevan los documentos a una representación adecuada para que finalmente se puedan aplicar distintos algoritmos de clasificación y así obtener el clasificador.

2.2 Representación de los Documentos

Para llevar a cabo la clasificación automática de texto se tiene que representar cada documento de los ejemplos de entrenamiento, de manera que a esa representación se le pueda aplicar el algoritmo de clasificación. La representación más utilizada es el modelo vectorial, ésta es manejada ampliamente por los sistemas de recuperación de información, además existen otras formas de representación de textos como los grafos, n-gramas, representación lógica, etc. Lo que se abordará en esta tesis es la representación vectorial, la cual se profundizará en los capítulos siguientes.

2.3 Stop Words

Las Stop words se define como términos que se consideran irrelevantes para la clasificación del documento, debido a que no presentan un contenido concreto que ayude al clasificador, además pueden aparecer con alta frecuencia en los documentos. Como ejemplos de este tipo de palabras son las preposiciones, conjunciones, artículos, etc., estas palabras se deben eliminar del documento para ayudar a clasificar mejor.

2.4 Stemming

El uso de lematizador es otra estrategia para eliminar afijos de una palabra, para poder representar palabras que significan lo mismo pero se escriben diferente ya sea por su conjugación o algún otro motivo, por ejemplo, caminar, caminará, caminó, caminando, todo este concepto significa una acción por lo cual se representará por camin.

2.5 Algoritmo de Aprendizaje

En la clasificación automática de textos se parte de una serie de clases o categorías prediseñadas, en las cuales hay que asignar cada uno de los documentos. E1 proceso de calcular patrones en base a los documentos preclasificados se conoce como entrenamiento [1], esos patrones se encuentran a través de los algoritmos de aprendizajes o las denominadas maquinas de aprendizaje. A continuación se presentan las maquinas de aprendizajes utilizados en esta investigación:

2.5.1 Naive Bayes

Uno de los algoritmos más utilizados para el cálculo de estos patrones es Naive Bayes, este es un clasificador probabilístico, el cual se basa en el cálculo de distribuciones de probabilidad en función a los datos observados.

El algoritmo necesita como entrada un documento que se denomina d y un conjunto de clases predefinidas $\{c_1, c_2, \ldots, c_k\}$, el clasificador Naive Bayes primero calcula la probabilidad a posteriori de que el documento pertenezca a cada clase particular c_k, es decir, $P(c_k|d)$ y entonces asigna el documento a la clase o clases con las probabilidades más altas. La probabilidad a posteriori es calculada aplicando el teorema de Bayes:

$$P(c_k/d) = \frac{P(d|c_k)P(c_k)}{P(d)} \quad (2.5.1)$$

El denominador P(d) en la formula es independiente de las clases; por lo cual tanto, puede ser ignorado. Por ende:

$$P(c_k|d) = P(d|c_k)P(c_k) \quad (2.5.2)$$

En Naive Bayes, se asume independencia de ocurrencia entre los términos del vocabulario c_k, es decir, que se puede calcular $P(c_k|d)$ a través de la formula a partir de la productora especificada en [1].

Uno de los problemas que presenta este método de aprendizaje es el de la probabilidad cero, para solucionar se realiza la estimación usando Laplace (agrega uno). Con esta estimación se pretende que todas las posibles incluidas las no vistas, tengan una probabilidad asociada, ya que la configuración que no esté en la colección de datos tendrá una probabilidad cero.

La tarea de aprendizaje con Naive Bayes consiste en construir una hipótesis por medio de estimar las probabilidades $P(c_k)$ y $P(t_i|c_k)$ en términos de los ejemplos de entrenamiento pertenecientes a la clase c_k.

2.5.2 Máquinas de Soporte Vectorial (SVM)

El estándar de SVM toma un conjunto de datos de entrada y predice en cuál de las clases posibles comprende la entrada. Este clasificador se cataloga no probabilístico pero sí lineal y binario. Un modelo de SVM es una representación de los ejemplos como puntos en el espacio, mapeados de manera que los ejemplos de las categorías están divididas o separadas por una brecha de espacio, tan amplio como sea posible, nuevos ejemplos se asignas en el plano y se prevé que pertenecen a una categoría basado en qué lado de la brecha son graficados.

La máquina de soporte vectorial, formalmente construye un hiperplano o conjuntos de hiperplanos en un especio de dimensión infinita, que puede ser utilizado para la clasificación, regresión, u otras tareas. Una buena separación se logra por el hiperplano que tiene la mayor distancia a los puntos de los datos de entrenamiento, ya que en general cuanto mayor sea el margen funcional, más bajo es el error de generalización del clasificador. A menudo sucede que los conjuntos a discriminar no son separables linealmente en el espacio de dimensión

infinita, por esta razón se propuso que el espacio de dimensión finita original en el cual se mapea en un espacio mucho mayor en dimensiones, haciendo presumiblemente más fácil la separación en ese espacio más amplio.

El LIBSVM [4] es para una estimación multiclase, apoyando con la probabilidad el aprendizaje, basado en escala para valores de confianza adecuados, después de aplicar el modelo de aprendizaje en un conjunto de datos de clasificación.

2.5.3 K-NN

El algoritmo de K vecinos más próximos se basa en el aprendizaje por analogía, es decir, mediante la comparación de un ejemplo de ensayo dado con ejemplos de entrenamiento que son similares a la misma. Los ejemplos de entrenamiento son descritos por n atributos, cada ejemplo representa un punto en un espacio n-dimensional, de esta manera todos los ejemplos de entrenamiento se almacenan en un espacio de patrones n- dimensional. Cuando se da un ejemplo desconocido, el algoritmo busca el espacio de patrones de los ejemplos de entrenamiento que están más cerca del ejemplo desconocido. Cercano se define en términos de una distancia métrica, tal como la distancia euclidiana.

El algoritmo K-NN está entre el más simple de todos los algoritmos de aprendizaje automático. Los vecinos se han tomados del conjunto de ejemplos que han sido clasificados previamente por un experto, especificando su correcta clasificación. Esto se puede considerar como el conjunto de entrenamiento para el algoritmo.

El algoritmo básico K vecinos más próximos se compone de dos pasos: Encontrar los ejemplos de entrenamiento k que están más cerca el ejemplo no se ve y como segundo paso se toma la clasificación más común que ocurre en estos ejemplos k.

2.5.4 Decision Tree

Un árbol de decisión es un gráfico o modelo en forma de árbol, mas gráficamente es un árbol invertido, ya que tiene su raíz en la parte superior y crece hacia abajo. Esta representación de los datos tiene la ventaja, en comparación con otros enfoques, de ser significativo y fácil de interpretar. El objetivo es crear un modelo de clasificación que predice el valor de un atributo de destino (a menudo llamado clase o etiqueta) basado en varios atributos de entrada del conjunto ejemplo o de entrenamiento. Cada nodo interior del árbol corresponde a uno de los atributos de entrada. El número de bordes de un nodo interior nominal es igual al número de posibles valores del atributo de entrada correspondiente. Bordes salientes de atributos numéricos se etiquetan con rangos disjuntos. Cada nodo hoja representa un valor del atributo de la etiqueta dados los valores de los atributos de entrada representados por el camino desde la raíz a la hoja.

Árboles de decisión son generados por el particionamiento recursivo. Particionamiento recursivo significa dividir en varias ocasiones en los valores de los atributos. En cada recursión del algoritmo sigue los siguientes pasos:

Un atributo A se selecciona para dividir. Hacer una buena selección de atributos para dividir en cada etapa es crucial para la generación de un árbol útil. Los ejemplos de entrenamiento están ordenados en subconjuntos, uno para cada valor del atributo A en caso de un atributo nominal. En caso de atributos numéricos, se forman subconjuntos disjuntos para rangos de valores de atributo.

Un árbol se devuelve con un borde o una rama para cada subgrupo. Cada rama tiene un subárbol descendiente o un valor de etiqueta producida aplicando el mismo algoritmo de forma recursiva.

En general, la recursión se detiene cuando todos los ejemplos o instancias tienen el mismo valor de la etiqueta, es decir, el subconjunto es puro. O recursividad puede detenerse si la mayoría de los ejemplos son del mismo valor de la etiqueta. Esta es una generalización de la primera aproximación; con algún umbral de error. Sin embargo, hay otras condiciones vacilantes como:

La poda es una técnica en la que se eliminan los nodos hoja que no se suman a la capacidad discriminativa del árbol de decisión. Esto se hace para convertir un árbol demasiado específico o sobre equipado de una forma más general, a fin de mejorar su capacidad de predicción en los conjuntos de datos que no se ven. Pre-poda es un tipo de poda paralelo realizado con el proceso de creación del árbol. Post-poda, por otro lado, se hace después de que el proceso de creación del árbol es completo.

2.5.5 SMO

Implementa el algoritmo secuencial de optimización mínima de John Platt para el entrenamiento de un clasificador de vectores soporte. Esta aplicación sustituye a nivel general todos los valores y transforma atributos nominales en binarios. También normaliza todos los atributos por defecto. Problemas Multi-clase se resuelven utilizando la clasificación por parejas (1 - vs- 1 y si logística modelos se construyen de acoplamiento por pares de acuerdo con Hastie y Tibshirani, 1.998). Para obtener estimaciones de probabilidad apropiadas, utiliza la opción que se ajuste a los modelos de regresión logística para las salidas de la máquina de vectores de soporte. En el caso multi-clase las probabilidades predichas se acoplan mediante Hastie y método de acoplamiento por parejas de Tibshirani.

2.6 Medidas de Evaluación

Para evaluar un sistema de clasificación de texto se utilizan las medidas de precisión y recuerdo (precisión π and recall ρ), que son medidas comunes en el área de recuperación de información, además para evaluar el desempeño se utiliza F-medida (F-measure metric).

2.6.1 Precisión y Recuerdo:

La precisión es la probabilidad de que un documento clasificado en la clase corresponda realmente a esa clase. El recuerdo es la probabilidad de que un documento que pertenece a la clase es clasificado dentro de esa clase [5]. Así la precisión se puede ver como una medida de la corrección del sistema, mientras que el recuerdo da una medida de cobertura o completitud.

Tabla 2-1 Matriz de Confusión

		Actual Value	
		Positives	Negatives
Predited Value	Positives	TP (True Positive)	FP (False Positive)
	Negatives	FN (False Negative)	TN (True Negative)

En la tabla 3.3.1, se presenta TP (verdaderos positivos) es el número de documentos asignados correctamente a la clase, FP (falsos positivos) es el número de documentos que no pertenecen a la clase, pero se asignan a la clase incorrectamente por el clasificador, FN (falsos negativos) es el número de documentos que no son asignados por la clase por el clasificador, pero que en realidad pertenecen a la clase, TN (verdaderos negativos) es el número de documentos asignados correctamente a la clase negativa.

Los valores de la tabla permiten estimar las medidas de precisión y recuerdo según las siguientes expresiones:

$$\pi_i = \frac{\mathrm{TP}_i}{\mathrm{TP}_i + \mathrm{FP}_i} \quad (2.6.1)$$

La precisión expresa en qué medida de clasificador toma la decisión correcta al ubicar cualquier documento en la clase que le corresponde.

$$\rho_i = \frac{\mathrm{TP}_i}{\mathrm{TP}_i + \mathrm{FN}_i} \quad (2.6.2)$$

El recuerdo refleja cuantos de todos los documentos de una clase son clasificados en ella.

2.6.2 F-Medida:

Los valores de F-medida se mueve en un intervalo de (0,1) y los valores de mas altos corresponden a una mayor claridad del clasificador. La puntuación del F-medida de problemas de clasificación de múltiples clases se puede calcular por dos formas: medida-micro (micro-average) y medida-macro (macro-averange) [10].

2.6.2.1 Micro-Promedio de F-Medida:

Se calcula a nivel global de todas las decisiones de cada categoría. ρ y π se obtiene mediante la suma de todas las decisiones individuales:

$$\pi = \frac{TP}{TP+FP} = \frac{\sum_{i=1}^{M} TP_i}{\sum_{i=1}^{M}(TP_i+FP_i)}, \quad (2.6.3)$$

$$\rho = \frac{TP}{TP+FN} = \frac{\sum_{i=1}^{M} TP_i}{\sum_{i=1}^{M}(TP_i+FN_i)} \quad (2.6.4)$$

Donde M es el número de categorías. Micro-Promedio F-Medida se calcula como:

$$F(micro-averanged) = \frac{2\pi\rho}{\pi+\rho} \quad (2.6.5)$$

Da la misma importancia a cada documento y, por tanto, se considera como la media aritmética de todos los documentos/categoría pares. Tiende a ser dominada por el rendimiento del clasificador en categorías comunes.

2.6.2.2 Macro-Promedio de F-Medida:

Se calcula localmente sobre cada categoría y luego se toma la media aritmética de todas las categorías. π y ρ se calculan para cada categoría como se presenta a continuación, donde F-Medida para cada categoría i se calcula tomando el promedio de los valores de F-medida para cada categoría:

$$F_i = \frac{2\pi_i\rho_i}{\pi_i+\rho_i}, \quad F(macro-averaged) = \frac{\sum_{i=1}^{M} F_i}{M} \quad (2.6.6)$$

Donde M es el número total de categorías. Macro-Promedio de F-Medida da igual importancia para cada categoría, independiente de su frecuencia. Está influenciada más por el rendimiento del clasificador en categorías poco comunes.

2.6.3.1 Acierto:

Otra medida que es empleada en este trabajo de tesis es la exactitud, la cual representa el porcentaje de las predicciones que son correctas.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN} \quad (2.6.7)$$

3. Estado del arte

La clasificación automática de textos depende en gran medida de la representación de los datos, para lograr el objetivo de clasificar. A continuación se presentan formas de representar textos investigados para las pruebas desarrolladas en esta investigación, se toma tan solo la representación vectorial pero, cabe mencionar que también existen más tipos de representaciones distinta a vectores.

3.1 Modelo Vectorial

Este modelo consiste en representar la colección de documentos como una matriz de palabras o términos por documentos [6]. Cada texto o documento es representado por medio de un vector de términos de los cuales los términos son las palabras del texto. Frecuentemente, el conjunto de palabras del vector, es el resultado de filtrar las palabras del vocabulario con respecto a una lista de palabras vacías, se llama vacías por no tener información semántica, como por ejemplo las preposiciones, conjunciones, artículos, etc.

Los documentos contienen símbolos o palabras las cuales necesitan un análisis profundo del contenido lingüístico, podemos elegir para describir cada documento por características que representan es decir por tokens frecuentes. El conjunto de características colectiva se suele llamar un diccionario. Los símbolos o palabras en el diccionario son la base para la creación de datos numéricos que corresponden a la colección de documentos. Cada fila es un documento, y cada columna representa una función. Por lo tanto, una celda en la hoja de cálculo es una medida de una característica (correspondiente a la columna) para un documento (que corresponde a una fila) [7].

3.1.1 Vectores Binarios

En el modelo más básico de estos datos, simplemente comprobar la presencia o ausencia de las palabras, y la celda de entradas son binarias correspondientes a un documento y una palabra. El diccionario de palabras cubre todas las posibilidades y corresponde para el número de columnas de la hoja de cálculo. Las celdas tendrán unos o ceros, dependiendo de si se encuentran las palabras el documento. En muchas circunstancias, es posible que desee trabajar con un pequeño diccionario. La muestra puede ser relativamente pequeña, o un diccionario de gran tamaño puede ser difícil de manejar. En tales casos, podríamos tratar de reducir el tamaño del diccionario por varias transformaciones. Como la predicción es el objetivo, necesitamos una o más columnas para la correcta Respuesta (o clase) para cada documento.

3.1.2 Vectores de Frecuencia

Información de la frecuencia de la palabra cuenta puede ser muy útil en la reducción de tamaño del diccionario y, a veces puede mejorar la predicción de algunos métodos. Las palabras más frecuentes suelen ser palabras vacías y puede ser eliminado. El resto de las palabras de uso más frecuente son a menudo las palabras importantes que deben permanecer en un diccionario local. Las palabras son muy raras veces los errores tipográficos y pueden ser descartadas. Para algunos métodos de aprendizaje, un diccionario local de las palabras más frecuentes, quizás menos de 200, puede ser sorprendentemente eficaz. Un enfoque alternativo para la generación de diccionario local es para generar un diccionario mundial de todos los documentos de la colección. Característica rutinas de selección tratará de seleccionar un subconjunto de las palabras que aparecen tener el mayor potencial para la predicción. Estos métodos de selección son a menudo complicados e independientes del método de predicción. En general no se usa y depende sólo información de la frecuencia, que es bastante fácil de determinar [7].

En esta investigación se prueba con 2 tipos de frecuencia, las cuales se describen a continuación:

3.1.2.1 Ocurrencia o Frecuencia Absoluta

Cada columna en la hoja de cálculo corresponde a una característica. En lugar de ceros o unos como entradas en las celdas de la hoja de cálculo, la frecuencia real de ocurrencia de la palabra puede se expresa en cada celda. Si una palabra aparece diez veces en un documento, este conteo se introduce en la celda. Para algunos de los métodos de aprendizaje, el recuento da un resultado ligeramente mejor.

3.1.2.2 Frecuencia o Frecuencia Relativa

Cada columna de la hoja de cálculo corresponde a un valor, en vez de ser la frecuencia absoluta u ocurrencia, es una proporción de una palabra en un documento y se calcula como:

$$tf_{ij} = \frac{w_{ij}}{\sum_k w_{kj}} \quad (3.1.1)$$

Se dice que la frecuencia es relativa al número de palabras en un documento y puede ser engañosa dependiendo de la longitud del documento.

3.1.3 Vectores de importancia

El siguiente paso más allá de contar la frecuencia de una palabra en un documento es modificar el recuento por la importancia percibida de esa palabra. La importancia de una palabra en un documento se representa por los siguientes métodos:

Tabla 3-1 Representación de datos en la dataset

	t	$\bar{t}$
Etiqueta $_1$	a_{t,λ_1}	d_{t,λ_1}
Etiqueta $_\lambda$	a_{t,λ_j}	d_{t,λ_j}
Etiqueta $_{\|L\|}$	$a_{t,\|L\|}$	$d_{t,\|L\|}$

3.1.3.1 Representación a través Bolsas de palabras o tf-idf (Bag-of-Words represetation)

La representación de documentos más utilizados para la clasificación de textos tf-idf [3], se ha utilizado para calcular las ponderaciones o calificaciones de palabras. Cada componente del vector se calcula como:

$$tf - idf_{td} = f_{t,d} \log_{10}(\frac{N}{N_t}), \text{(3.1.2)}$$

Donde $f_{t,d}$es la frecuencia del término t en el documento $d, N = \left(a_{t,\lambda_1} + d_{t,\lambda_1} + a_{t,\lambda_2} + d_{t,\lambda_2}\right)$ es el número de documentos, y $N_t = \left(a_{t,\lambda_1} + a_{t,\lambda_2}\right)$ es el número de documentos que contienen el término t.

Se logra apreciar que el peso tf-idf asigna a la palabra *j* es la frecuencia del término (es decir, el número de palabras) modificado por un factor de escala por la importancia de la palabra. El factor de escala se llama la *inversa frecuencia de documento*, que se da en la ecuación (4.2). Simplemente comprueba el número de documentos que contienen la palabra *j* (es decir, *df* (*j*)) y revierte la escala. Por lo tanto, cuando una palabra aparece en muchos documentos, es considerado de importancia y la escala se reduce, tal vez cerca de cero. ¿Cuándo la palabra es relativamente única y aparece en algunos documentos, la escala factor de zoom hacia arriba, ya que parece importante.

$$\text{tf-idf}(j) = \text{tf}(j) * \text{idf}(j) . \text{(3.1.3)}$$

$$\text{idf}(j) = \log(N/\text{df}(j)). \text{(3.1.4)}$$

3.1.3.2 Representación de la relevancia de la frecuencia o tf-rf (relevance frequency representation):

Propuesto recientemente por Lan et al. [10], como una representación VSM mejorando el basado en dos clases y los problemas de etiqueta única. Cada componente del vector se calcula:

$$tf - rf_{td} = f_{t,d} \log_2(2 + \frac{a_{t,\lambda_1}}{\max(1, a_{t,\lambda_2})}), \quad (3.1.5)$$

Donde $f_{t,d}$es la frecuencia del término t en el documento d, la expresión a_{t,λ_1}es en número de documentos en la clase positiva que contiene el término t y la expresión a_{t,λ_2}es el número de documentos en la clase negativa que contiene el término t. La función $\max(1, a_{t,\lambda_2})$ en el denominador permite que el término $tf - rf_{td}$ no llegue a ser indefinido por si a_{t,λ_2} toma el valor de cero.

En el caso de la calcificación de clases múltiple, se utiliza un método de uno contra todos. Hay que tomar en cuenta que la representación tf-rf es para problemas de etiqueta única y no considera la información de la frecuencia del término evaluado en otras clases. Es decir, que sólo tiene en cuenta la relación de la aparición del término en la clase bajo evaluación (es decir, la positiva) frente a todas las otras clases (es decir, las negativas).

3.2 Nuevas propuestas

La propuesta que se plantea en [11], explica que desde el punto de vista teórico es una aplicación de la representación de textos tf-rf, ya que cambia la representación de un documento de acuerdo con la etiqueta de objeto de evaluación, lo que se consigue mayor diferencias entre los documentos que pertenecen a diferentes etiquetas y por lo tanto el aprovechamiento de el rendimiento de los clasificadores binarios. Por lo tanto, la información importante acerca de la que se utiliza en la frecuencia en otras clases, especialmente cuando la frecuencia de los términos de las celdas sufren variaciones, [11] propone el uso de una función de centralidad μ-Relevancia de frecuencia para cada etiqueta, se deriva de la frecuencia de los términos y la frecuencia pertinente de una determinada etiqueta, constituye una nueva representación basado en tf-rf para un problema con varias etiquetas:

$$tf - urfl_{tdl} = f_{t,d} \log_2\left(2 + \frac{a_{t,l}}{\mu\left(a_{t,\lambda_{j/t}}\right)}\right), \quad (3.2.1)$$

Donde $\mu\left(a_{t,\lambda_{j/t}}\right)$ es la función sobre el set $a_{t,\lambda_{j/t}} = \{a_{t,\lambda_1}, \dots, a_{t,\lambda_{l-1}}, a_{t,\lambda_{l+1}}, \dots, a_{t,\lambda_{l-|L|}}\}$. Considerando $\mu\left(a_{t,\lambda_{j/l}}\right) = \max\left(1, mean\left(a_{t,\lambda_l}\right)\right)$, para tf-rfl representación y $\mu\left(a_{t,\lambda_{j/l}}\right) =$

$\max\left(1, mean\left(a_{t,\lambda_l}\right)\right)$, para tf-rrfl representación. Tal métrica clásica y la mediana es una métrica robusta.

3.2.1 Relevancia de la frecuencia de una etiqueta:

La relevancia de la frecuencia de un etiqueta, tf-rfl, se deriva de la μ-Relevancia frecuencia de una etiqueta, tf-μrfl, como tal, constituye una nueva representación para un problema de múltiples etiqueta.

$$tf - rfl_{tdl} = f_{t,d} \log_2\left(2 + \frac{a_{t,l}}{\max\left(1, mean\left(a_{t,\lambda_{j/l}}\right)\right)}\right), \quad (3.2.2)$$

En la ecuación, el término $mean\left(a_{t,\lambda_{j/l}}\right)$ es el número medio de los documentos que contienen el término t para cada documento etiquetado distinto a l.

3.2.2 Frecuencia robusta de la relevancia de una etiqueta:

Frecuencia robusta de la relevancia de una etiqueta, tf-rrfl, también deriva de la μ-Relevancia de la frecuencia de una etiqueta, tf-μrfl y como tal, esta es la segunda nueva representación de un problema con varias etiquetas.

$$tf - rrfl_{tdl} = f_{t,d} \log_2\left(2 + \frac{a_{t,l}}{\max\left(1, median\left(a_{t,\lambda_{j/l}}\right)\right)}\right), \quad (3.2.3)$$

El uso de la mediana debe producir resultados más sólidos en los conjuntos de datos que contienen grandes diferencias entre la frecuencia de la ocurrencia de un término en un determinado conjunto de etiquetas frente a otros conjuntos de etiquetas bajo evaluación.

Existen versiones alternativas de la formulación básica tf-idf, pero la motivación general es la misma. El resultado neto de este proceso es una puntuación positiva que sustituye a la frecuencia simple o binaria de verdadero o falso entrada en la celda de de nuestra hoja de cálculo. Cuanto mayor es la puntuación, más importante es su valor esperado para el método de aprendizaje. Aunque esta transformación es sólo una ligera modificación del modelo binario-rasgo original, que no pierden la claridad y simplicidad de la presentación anterior [7].

4. Definición del problema

El problema se define en: propiciar a una representación de textos con un buen desempeño en la clasificación de textos cortos, por medios de algoritmos de aprendizaje artificial.

El propósito de la investigación es comparar los métodos de representación de textos, para poder distinguir el desempeño de estos con los diferentes algoritmos de inteligencia artificial, logrando la clasificación de estos. Las métricas a utilizar son las que se nombraron en capítulos anteriores son: la precisión, el recuerdo, exactitud, F-Medida.

4.1 Solución Propuesta

La solución propuesta para poder evaluar el desempeño de cada representación, tomando las representaciones vistas en el capítulo 3.0 a través de, las distintas formas de clasificar con los algoritmo de aprendizaje automático analizados en la sección 2.5 y que nos brinda la herramienta de minería de datos, para que de esta manera poder discernir entre cual método es mejor para representar y con qué método de clasificación.

4.2 Implementación de solución

La implementación se desarrolló en la herramienta de aprendizaje automático y minería de datos "RapidMiner", [8] el cual permite el desarrollo de procesos de análisis de datos mediante el encadenamiento de operadores a través de un entorno gráfico. Se usa en investigación educación, capacitación, creación rápida de prototipos y en aplicaciones empresariales. En una encuesta realizada por un periódico de minería de datos [8], RapidMiner ocupó el segundo lugar en herramientas de analítica y de minería de datos utilizadas para proyectos reales en 2009 y fue el primero en 2010.

Después de una larga búsqueda de herramientas para tratar datos de tipo texto, la más aceptable fue WEKA, pero como su interfaz es poco usable, la poca flexibilidad del ingreso de los tweets, se decidió trabajar con software libre, RapidMiner entre muchos otros que fueron analizados en el transcurso de proyecto, como Orange Cavas, el cual tiene una excelente interfaz, pero aunque agradable a la vista, es complicado para realizar lo que uno quiere, y de esta manera fueron descartadas JHepWork, KNIME. La característica que convenció por esta herramienta es que se pueden descargar complementos, entre ellos se encuentra el complemento de WEKA, con el cual se pueden ocupar características de esta herramienta.

4.3 Presentación de los Dataset

Los mensajes de Twitter utilizados en esta investigación son extraídos de la investigación de [9], en esta se clasifican los Twitter según intención del usuario. A continuación se presentan las categorías identificadas, se identifican ocho categorías que

incluyen las intenciones más comunes de los usuarios en los mensajes y que son relevantes para el estudio.

- Reporte Noticia
- Opinión Noticia
- Publicidad
- Opinión General
- Compartir Ubicación / Evento
- Chat
- Pregunta
- Mensaje Personal

A continuación se presenta una descripción detallada de las categorías que se han identificado de la intención de los usuarios en los mensajes de Twitter o bien conocidos como Tweets, el objetivo es definirlos para saber cómo van a ser considerados en cada categoría.

4.3.1 Reporte de Noticia (RN)

Un reporte de notica corresponde a una noticia emitida de manera objetiva. Por lo general emitida por una cuenta corporativa (CNN, Cooperativa, Terra, etc.) y acompañada de un hipervínculo a la noticia completa en el sitio web del informante. En este caso el Tweet se encuentra escrito en lenguaje formal.

Tabla 4-1 Ejemplo dataset RN

Emisor	Mensaje
Cooperativa	Terremoto en Turquía alcanzó los 6 grados Richter http://bit.ly/IUOUnL
Cooperativa	Peñarol acabó con el sueño de U. Católica en la Copa Libertadores http://bit.ly/mvlix9

4.3.2 Opinión de Noticia (ON)

A diferencia del Reporte de Noticia que es un mensaje objetivo, en la Opinión de Noticia se emite comentario, ya sea positivo o negativo, relativo al tema de la noticia. Esta no debería provenir de una cuenta corporativa ya que las noticias emitidas por estos son realizadas de manera objetiva.

Tabla 4-2 Ejemplo dataset ON

Emisor	Mensaje
EdxBigBro	RT @Cooperativa: RN llamó al Gobierno a “aplicar mano dura” a quienes causen destrozos el 21 de mayo http://bit.ly/kJIQ0k // como siempre!

4.3.3 Publicidad (PU)

Al igual que los Reportes de Noticia estos provienen de una cuenta corporativa. Usualmente podrían ser considerados spam, debido a que incluyen un hipervínculo al sitio de la oferta y palabras que hacen referencia a esta (ej. Descuento, gratis, promoción, etc.).

Tabla 4-3 Ejemplo dataset PU

Emisor	Mensaje
Santanderchile	Obtén 10%dcto. En la tasa al contratar tu Crédito de Consumo en Santander.cl http://bit.ly/jiLIst

4.3.4 Opinión General (OG)

Una opinión general representa el pensamiento del autor del Tweet sobre algún tema en particular. A diferencia de la opinión de notica, este usuario es quien comienza a opinar de un tema y no opinar sobre una noticia emitida a través de un tweet corporativo. Este mensaje suele estar escrito de manera más informar, acortando palabras, destacando sentimientos utilizando mayúsculas, haciendo uso de emoticones y otras técnicas que permitan demostrar opinión.

Tabla 4-4 Ejemplo dataset OG

Emisor	Mensaje
SoledadOnetto	Los del +56 etc me parece lo más latero. Esta noche no dormiré
Wefesx	Jajajaja la película del 13 fue muy malaa!! Lo único bueno fue ver a liv Tyler jajaja

4.3.5 Compartir Ubicación / Evento (CU)

Esta categoría hace referencia a compartir un mensaje que incluya información geográfica de dónde se encuentra el autor. Esto puede ser mediante el uso de servicios que permitan compartir la ubicación del usuario (ej. Foursquare) o informando de donde se encuentra sólo utilizando lenguaje natural (ej. “Me encuentro en…”, etc).

Tabla 4-5 Ejemplo data set CU

Emisor	Mensaje
Jorge_Galvez	En la onferencia de James Adams(@ Pontifica Universidad Católica de Valparaíso) http://4sq.com/IMROlw
manuguisone	En la conferencia de Enrique Schewach…superó mis expectativas

4.3.6 Chat (CH)

Un mensaje perteneciente a la categoría de chat representa la conversación entre uno o más usuarios de Twitter. Esta conversación se manifiesta a través del uso de “@” que permite nombrar a un usuario dentro de un mensaje. Por lo general en la conversación se nombra al o los usuarios al comienzo del mensaje.

Tabla 4-6 Ejemplo dataset CH

Emisor	Mensaje
Santanderchile	@giosalinas Debes solicitarlo en sucursal hasta las 14:00
mauciomartis	@alberto_holts esa es la idea. No está nada de mal el nombre. Evaluaré la posibilidad.

4.3.7 Pregunta (PR)

Dentro de esta categoría pueden aparecer dos tipos de preguntas generalmente realizadas. Pregunta directa, que se realizan a uno o más usuarios específicos haciendo uso de la posibilidad de nombrar a un usuario dentro del mensaje. Pregunta a los seguidores, corresponde a una pregunta abierta a todos los seguidores, esta no posee ningún destinatario especificado como en el caso de la pregunta directa.

Tabla 4-7 Ejemplo dataset PR

Emisor	Mensaje
santanderchile	@ng_cuevas podemos ayudarte en algo?
SoledadOnetto	Bajando whatsapp. Experiencias que deseen compartir?

4.3.8 Mensaje Personal (MP)

Un mensaje personal correspondiente a cualquier mensaje que sea compartido por un usuario de Twitter en el cual desee informar de asuntos personales y que no correspondan a ninguna de las otras categorías expuestas. Dentro de esta categoría se encuentran mensajes relacionados a la situación sentimental de la persona, información de lo que se encuentra haciendo, compartir algún tipo de anécdota, mantener presencia en la red, entre otros.

Tabla 4-8 Ejemplo dataset MP

Emisor	Mensaje
SoledadOneto	Parece el Valle de la Luna pero es Farellones antes del invierno blanco. Que paz! Yfrog.com/h4uj6rzj
SoledadOneto	Cerrada la oficina virtual. Abro mañana a las 8 am. Hora de ver “Hermanos” con Natalie Portman.
SoledadOneto	Viendo 40 y tantos…luego a dormir.

La manera de presentar los datos a RapidMiner es en carpetas, ya que esta aplicación tiene la característica de poder extraer datos de variados tipos de formatos, pero la manera en la cual el software debe realizar menos procesos es asignando a carpetas (files), cuyas carpetas deben tener los tweets de cada clase, en este caso de CH, CU, MP, OG, ON, PR, PU y RN; cada tweets debe estar en un formato .txt, de esta manera RapidMiner genera la data, string con su atributo.

Debido a que los dataset fueron entregados separados cada categoría en un solo archivo .txt, el cual separaba cada tweets solo por un salto de carro. Para lograr que estos tweets se separen distintos archivos .txt, fue necesario crear un programa en lenguaje C, el cual realiza la apertura del archivo fuente y crea los nuevos archivos. Se escogió C debió a su rápido procesamiento de los datos, así como el domino de este lenguaje de programación, como se puede observar en la imagen, el código del programa no supera las 33 líneas.

5. Pruebas Experimentales

En las pruebas realizadas, se utiliza la herramienta anteriormente descrita, ejecutada en un computador con sistema operativo Windows 7 Home Basic de 64 bits, con un procesador 4th Generation Intel® Core™ i7 Procesasor 15.6" Full HD (1920x1080), una menoría RAM de 36 GB, marca MSI.

Se realiza una validación cruzada con el fin de estimar la capacidad predictiva de un operador de aprendizaje. Se opta con este proceso debido a los bueno resultados que arroja.

El set de datos de entrada se divide en k subconjuntos de igual tamaño. De los subconjuntos de k, un solo subconjunto se mantiene como el conjunto de datos de prueba y el restante k-1 subconjuntos se utilizan como conjunto de datos de entrenamiento. A continuación, el proceso de validación cruzada se repite k veces, con cada uno de los subconjuntos K. Los k resultados de las iteraciones k se puede promediar para producir una única estimación. El valor de k se puede ajustar usando el número de parámetro de validaciones.

5.1 Pre procesamiento de datos

Antes de aplicar técnicas de selección de atributos se aplicaron pre procesos para reducir el número de atributos, esto se logra gracias a un plug-in para el procesamiento de texto que puede ser instalado en RapidMiner utilizando el servidor de actualizaciones buscando text minig.

Los archivos pueden ser transformados a la representación de la bolsa de palabras (create word vector). Los valores a usar como valor de los atributos de las palabra pueden ser seleccionados de entre representaciones explicadas en la sección 3.1 (TF-IDF, Term Frequency, Term Occurence, Binary Term ocurrence), pero no representa de las nuevas formas propuestas, para esto se ocupó un programa en Python, un lenguaje de programación multiplataforma y código abierto, para tomar la bases de datos de frecuencias y aplicar las formulas de representación de tf-rfl y tf-rrfl, junto con el plug-in para manipular estas bases de datos.

A continuación se presentan los operadores aplicados con Rapid Miner para realizar el pre procesamiento y el orden de aplicación respectivamente:

5.1.1 Función Tokenize de Rapid Miner

Este pre proceso toma todos los atributos que no son letras y los elimina de los documentos o tweet procesados, de esta manera la investigación se basa tan solo con palabras, aunque se sabe que los símbolos expresas sentimientos de los usuarios, pero este no es el motivo de la investigación.

5.1.2 Función Transform Case de Rapid Miner

Este operador se transforma todos los caracteres de un documento ya sea en minúsculas o mayúsculas, respectivamente. Para esta investigación se transformo todas las mayúsculas en minúsculas, de manera de que el software no diferencie palabras por estar escritas en mayúsculas o minúsculas, de esta manera se minimizar en gran medida los atributos.

5.1.3 Función Filter StopWords de Rapid Miner

Este operador filtra las palabras vacías que no tienen incidencia en la clasificación de textos porque no tienen significado propio, esta tarea es dependiente del lenguaje a filtrar. Este proceso eliminará palabras que son muy frecuentes en los documentos, pero que no tienen ninguna importancia para la tarea de aprendizaje (pronombres, preposiciones, adverbios, artículos, etc).

5.1.4 Función Filter Tokens (by Length) de Rapid Miner

Este operador filtra las palabras en función de longitud, es decir las elimina según la cantidad de caracteres que tiene, este filtro se aplico desde 3 letras, las palabras que tienen menos de 3 letras son eliminadas porque se consideran errores tipográficos.

5.1.5 Función Stem (Snowball) de Rapid Miner

Proceso aplicado antes de asignar valor a las celdas es la lematización (stemming). El problema con las palabras es que se puede tener atributos diferentes para la misma palabra, ya que pude ser escrita de diferentes maneras, por ejemplo, nombres en singular o plural, los verbos en diferentes tiempos, palabras masculinas o femeninas, prefijos, sufijos, etc., cada palabra se considera como un atributo diferente, la lematización (stemming) procesa las palabras y substituye todas las apariciones de la misma manera. Esto significa que el número de ocurrencias de cada palabra será más preciso.

5.2 Pruebas con Primer DataSet

La primera data tiene un volumen de 800 tweet, los cuales están divididos en grupos de 100 por cada clasificación, estos fueron extraídos por una función random realizada en Python. Los gráficos están divididos en las métricas anteriormente descritas, en la línea horizontal se presentan las 6 formas de representar textos estudiadas para esta investigación, y la línea vertical representa el porcentaje de la métrica asociada, los colores de las columnas representa los algoritmos de aprendizaje automáticos aplicados en esta investigación.

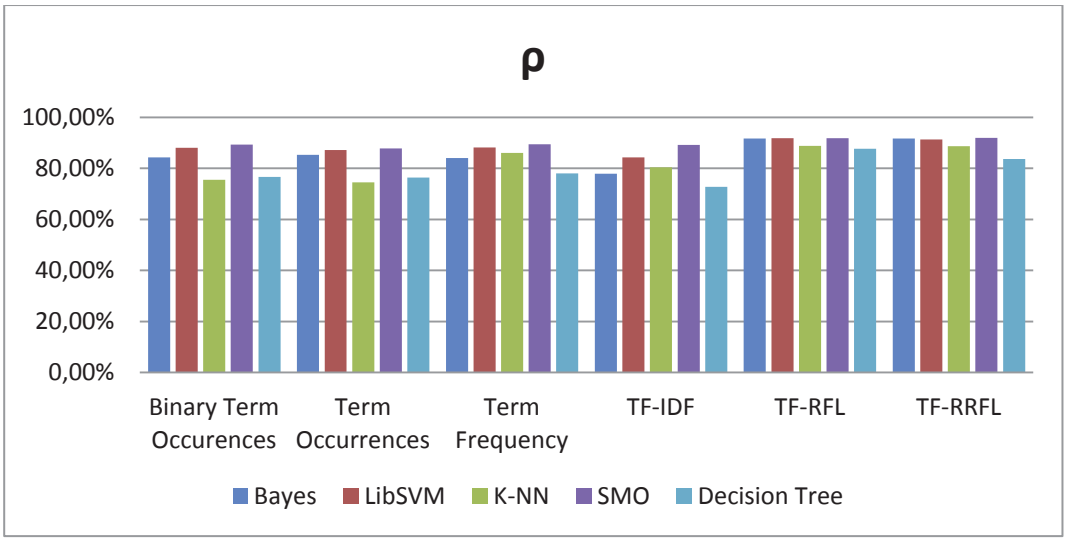

Figura 5-1 Gráfica recall primer data set

Se puede ver en el grafico que el mejor recall lo obtuvo SMO con la representación TF-RRFL, la cual obtuvo un recall de 91,91% promedio micro, es seguido de muy cerca por SVM con la representación TF-RFL, la cual muestra un recall de 91,84% promedio micro, en tercer lugar con un recall de 91,82% promedio micro lo obtuvo TF-RFL con el algoritmo SMO. En promedio entre todos los clasificadores en que arroja el mejor recall es TF-RRFL con un promedio de 90,39%.

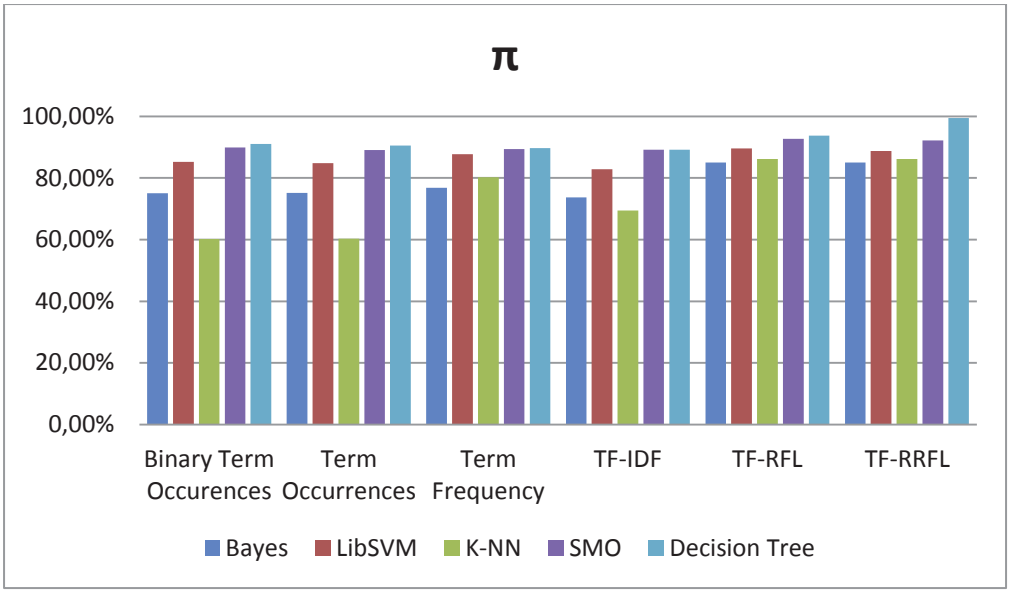

Figura 5-2 Gráfica precisión primer data set

En la grafica de precisión se puede observar que el método que obtuvo la mejor precisión fue Decision Tree con la representación TF-RRFL, obteniendo una precisión de 99,40% promedio micro, es seguido con un 93,78% promedio micro TF-RFL con el mismo algoritmo de aprendizaje, y en tercer lugar quien SMO con la presentación TF-RFL, con un 92,69% promedio micro. En promedio entre todos los clasificadores en que arroja la mejor precisión es TF-RRFL con un promedio de 90,30%.

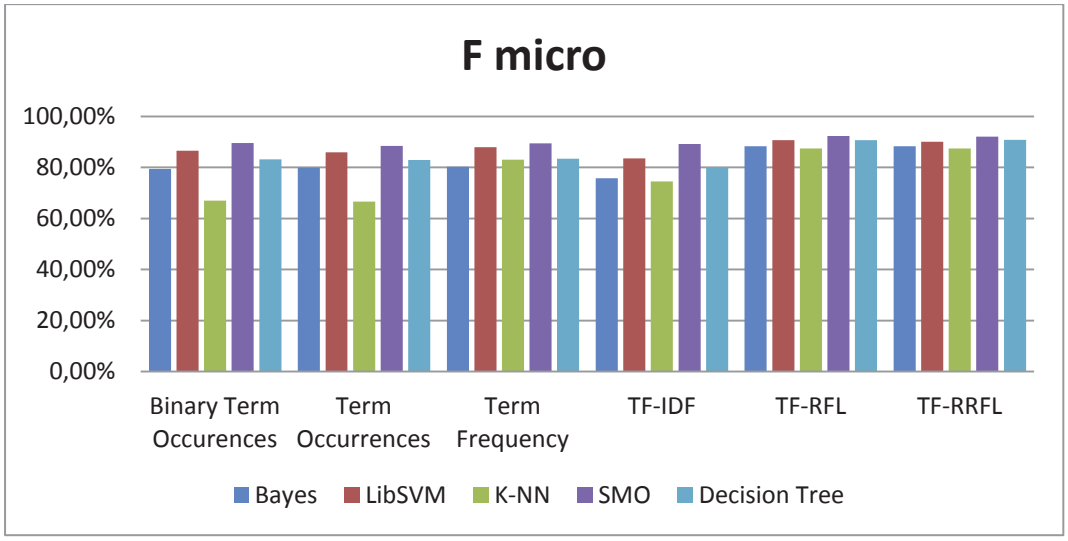

Figura 5-3 Gráfica F micro primer data set

En el grafico de F micro se puede ver que el mejor resultado lo arroja SMO con la representación TF-RFL con un f micro 92,26%, seguido muy de cerca por la representación TF-RRFL con el mismo clasificador da un f micro de 92,04%, y en tercer lugar con un 90,86% el clasificador Deision Tree con la representación TF-RRFL. En promedio entre todos los clasificadores en que arroja el mejor F micro es TF-RFL con un promedio de 89,87%.

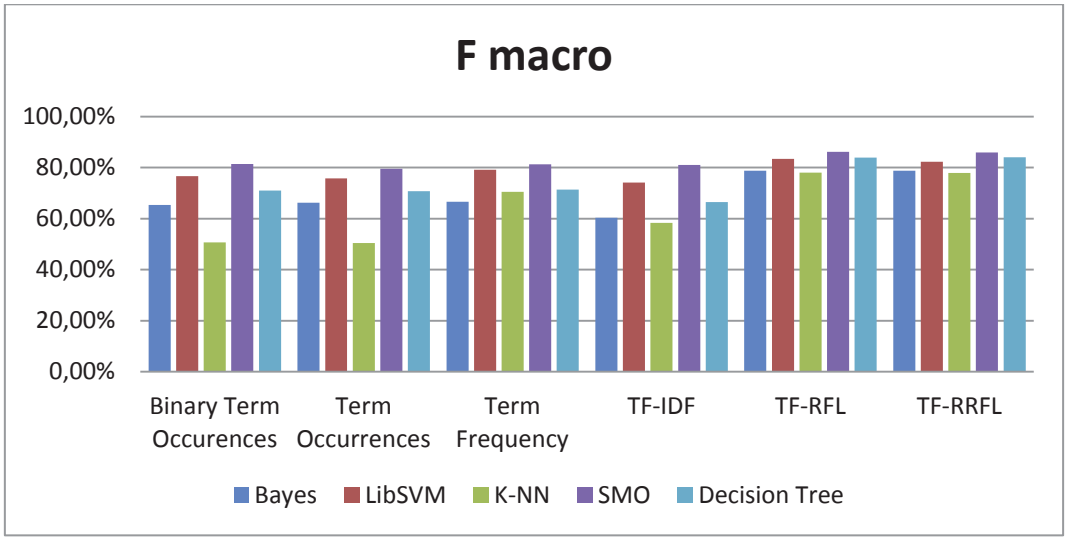

Figura 5-4 Gráfica F macro primer data set

El grafio de F macro se puede observar que el mejor resultado lo arroja SMO con la representación TF-RFL con un 86,20%, seguido por la representación TF-RRFL con el mismo clasificador se obtuvo un f maro de 85,87%, en tercer lugar se da a relucir Decision Tree con la representación TF-RRFL, con un 84,03%. En promedio entre todos los clasificadores en que arroja el mejor F macro es TF-RFL con un promedio de 80,57%.

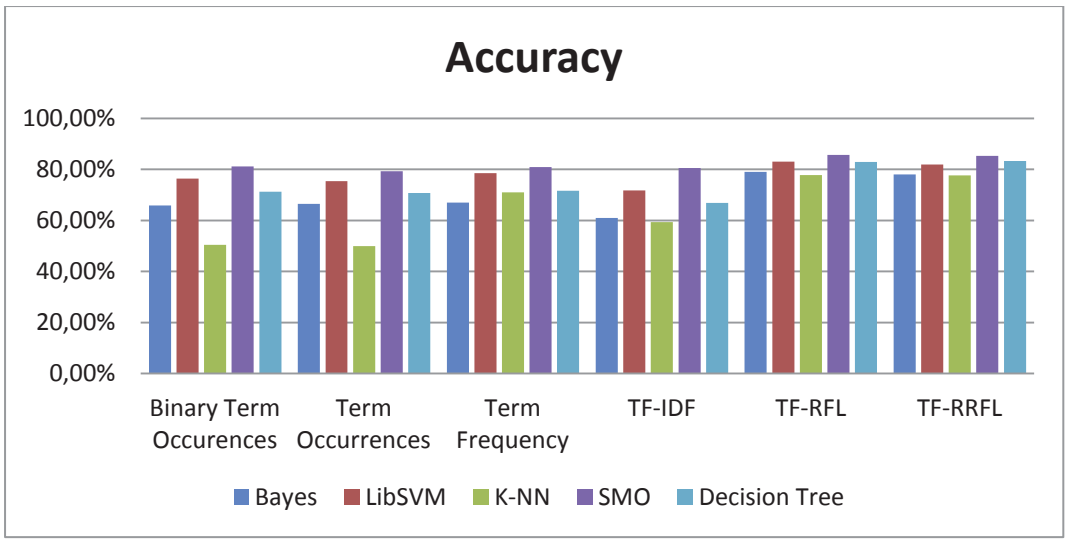

Figura 5-5 Grafica Accuracy primer data set

En el grafico de accuracy se ve reflejado que el mejor resultado fue por el clasificador SMO con la representación TF-RFL da un accuracy de 85,62%, seguido de cerca por la representación TF-RRFL con el mismo clasificador da un accuracy de 85,25%, en tercer lugar queda Decision Tree con la representación de TF-RRFL con un accuracy de 83,25%. En promedio entre todos los clasificadores en que arroja el mejor acierto es TF-RFL con un promedio de 81,65%.

5.3 Pruebas con Segundo DataSet

La segunda data tiene un volumen de 1501 tweet, los cuales están divididos en grupos de 200 por cada clasificación, con excepción de la clasificación de Opinión Noticia (ON), debido a que este contiene un total de 101 tweet de entrenamiento en esta clasificación. Los tweet fueron extraídos por una función randon realizada en Python. Los gráficos están divididos en las métricas anteriormente descritas, en la línea horizontal se presentan las 6 formas de representar textos estudiadas para esta investigación, y la línea vertical representa el porcentaje de la métrica asociada, los colores de las columnas representa los algoritmos de aprendizaje automáticos aplicados en esta investigación.

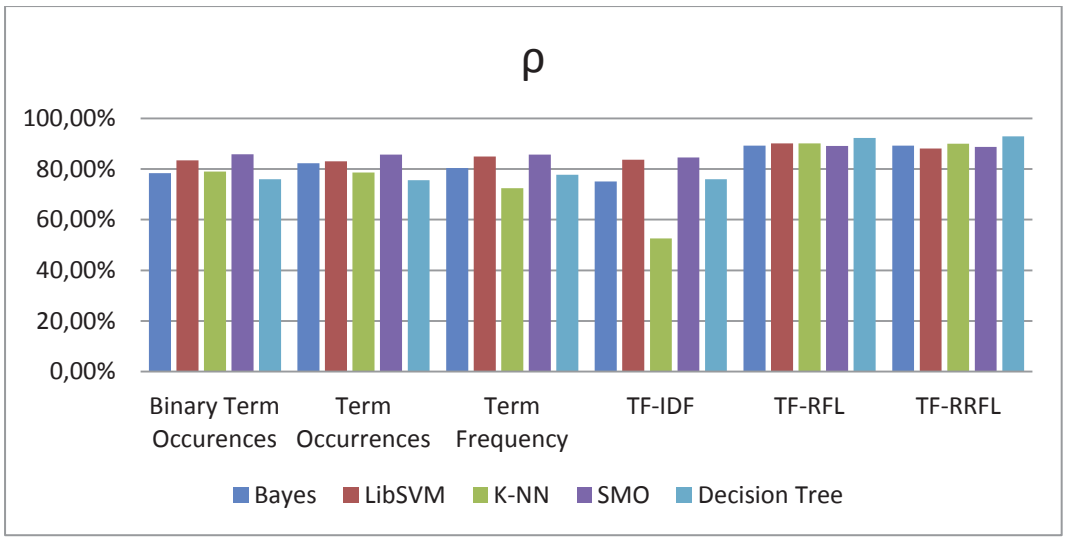

Figura 5-6 Gráfica recall segundo data set

En el grafico de recall se puede observar que el mejor resultado lo arroja Decision Tree con la representación TF-RRFL, con un recall de 92,85% promedio micro. El segundo mejor resultado lo arroja el mismo clasificador anterior pero con la representación TF-RFL, con un recall de 92,85% promedio micro. En tercer lugar queda el clasificador K-NN con la representación TF-RFL con un recall de 90,11% promedio micro. El mejor recall promediado entre los 5 clasificadores lo obtiene TF-RFL, con un promedio de 90,15%.

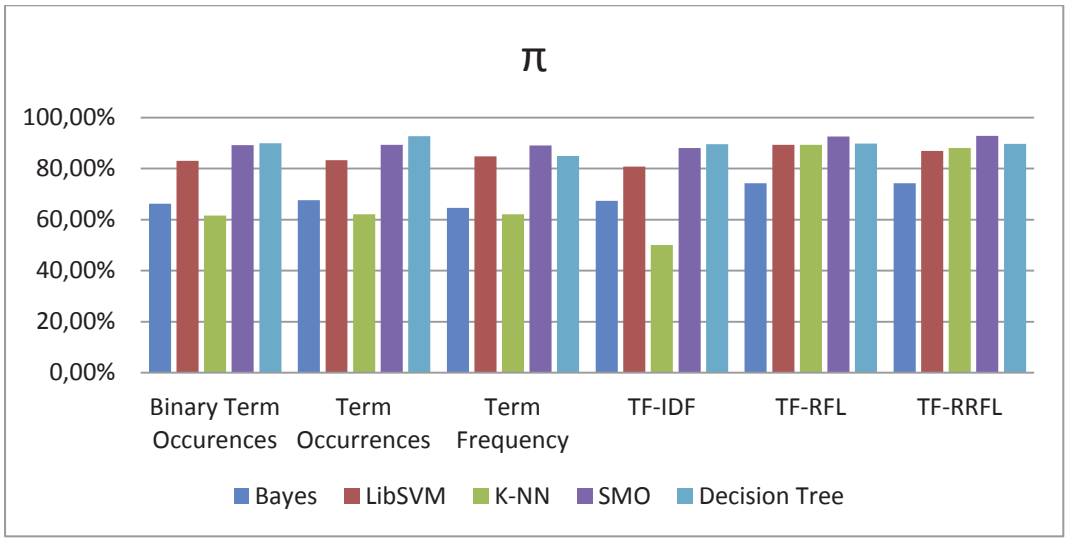

Figura 5-7 Gráfica precisión segundo data set

En el grafico anterior se ve representado la Precisión de los clasificadores, el mejor resultado lo arroja SMO con la representación TF-RRFL, con una precisión de 92,85% promedio micro. En segundo lugar queda el mismo clasificador pero con la representación TF-RFL, con una precisión de 92,59% promedio micro. En tercer lugar queda el clasificador

Decision Tree con la representación Term Occurrences, con una precisión de 92,68% promedio micro. En promedio, la representación que obtiene mejores resultados en cuanto a precisión es TF-RFL con un promedio de 87,04%.

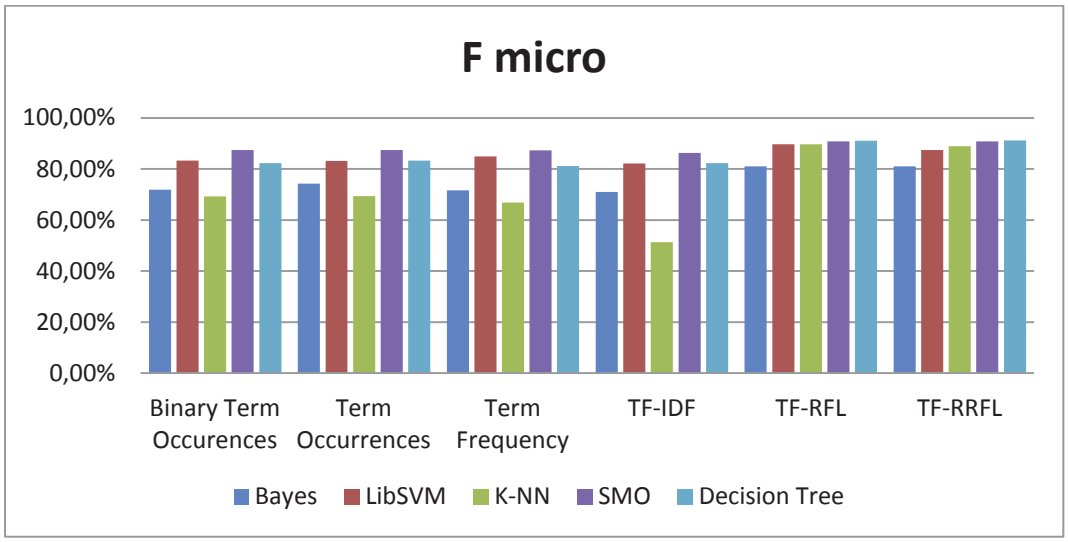

Figura 5-8 Gráfica F micro segundo data set

En el grafico anterior se presenta el F micro, en el cual se puede ver que el mejor resultado lo entrega Decision Tree con la representación TF-RRFL, con un F micro de 91,23%. En segundo lugar queda con un 90,99% el mismo clasificador anterior pero con la representación TF-RFL y en tercer lugar seguido de muy cerca por el clasificador SMO con la representación TF-RFL con un f micro de 90,84%. La representación que en promedio trabaja mejor con todos los clarificadores es TF-RFL con un promedio de 88,45%.

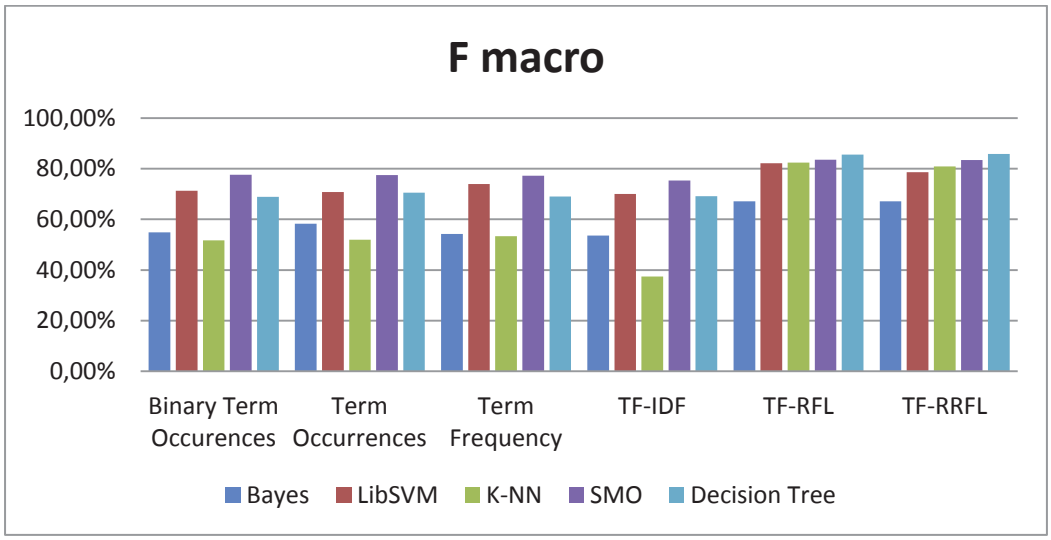

Figura 5-9 Gráfica F macro segundo data set

En el grafico anterior se presenta el F macro, en el cual se puede observar que el mejor resultado lo entrega Decision Tree con la representación TF-RRFL, con un F macro de 85,83%. En segundo lugar lo tiene el mismo clasificador pero con la representación TF-RFL con un F macro de 85,49% y en tercer lugar lo obtiene con un F macro de 83,48% SMO con la representación TF-RFL. En promedio el mejor resultado lo entrega TF-RFL con un promedio entre los clasificadores de 80,13%.

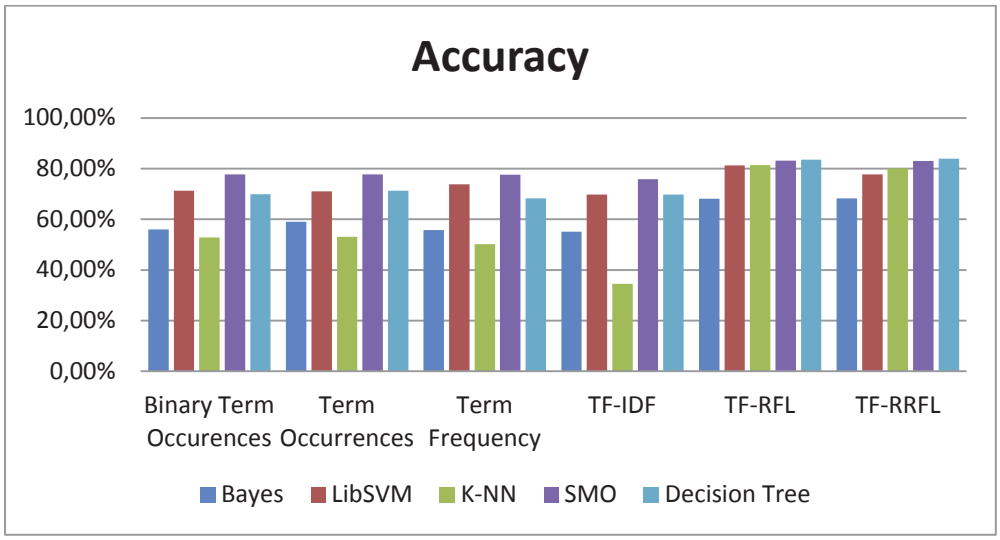

Figura 5-10 Gráfica Accuracy segundo data set

En el grafico anterior se presenta el acierto, en este se puede observar que el mejor acierto lo obtiene Decision Tree con la representación TF-RRFL con un 83,88%. En segundo lugar se observa al mismo clasificador pero con la representación TF-RFL, con un acierto del 83,48%. En tercer lugar se puede observar con un acierto del 83,21% a SMO con la representación TF-RFL. En promedio la presentación que obtiene un mejor acierto con los 5 clasificadores es TF-RFL con un promedio de acierto de 79,48%.

5.4 Pruebas con Data Set Completo

La tercera data tiene un volumen de 5239 tweet, los cuales fueron extraídos por una función randon realizada en Python. Los tweet están divididos en grupos dispersos, es decir no con la misma cantidad pro clasificación, la cual se presenta a continuación:

- 2427 Reporte Noticia (RN)
- 101 Opinión Noticia(ON)
- 417 Publicidad(PU)
- 321 Opinión General(OG)
- 659 Compartir Ubicación / Evento(CU)
- 600 Chat (CH)
- 339 Pregunta (PR)

- 375 Mensaje Personal (MP)

Los gráficos están divididos en las métricas anteriormente descritas, en la línea horizontal se presentan las 6 formas de representar textos estudiadas para esta investigación, y la línea vertical representa el porcentaje de la métrica asociada, los colores de las columnas representa los algoritmos de aprendizaje automáticos aplicados en esta investigación.

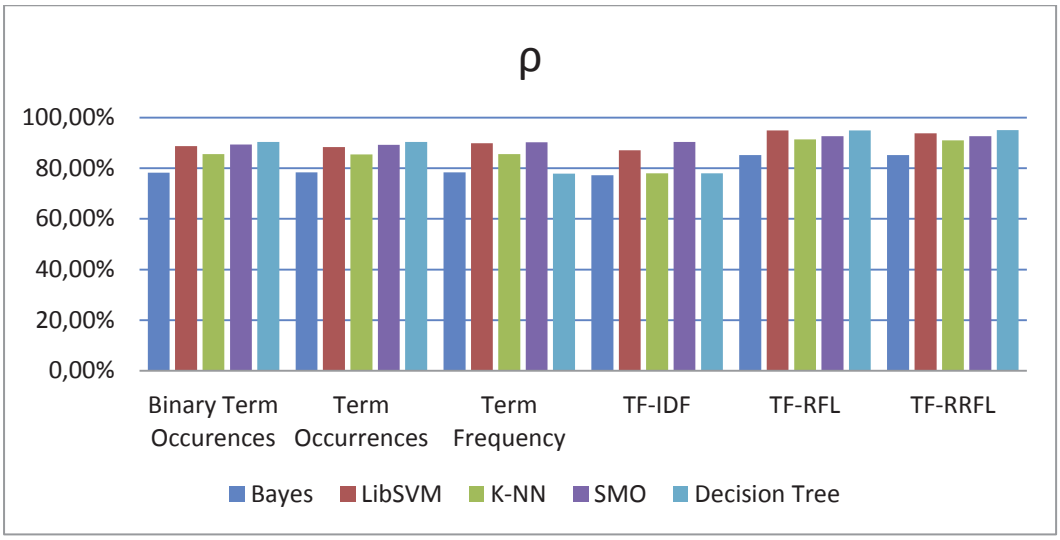

Figura 5-11 Gráfica recall tercer data set

El grafico anterior representa el recall de las pruebas aplicadas al data set, en este se puede observar que el mejor resultado lo arroja Decision Tree con la representación TF-RRFL con un recall de 95,05% promedio micro. En segundo lugar se lo queda el clasificador LibSVM con la representación TF-RFL con un recall de 94,89% promedio micro y esta seguido de muy cerca por el clasificador Decision Tree con la misma representación arroja un recall de 94,88% promedio micro. En promedio la representación que obtiene un mejor recall con los 5 clasificadores estudiados es TF-RFL con un promedio de recall de 90,27%.

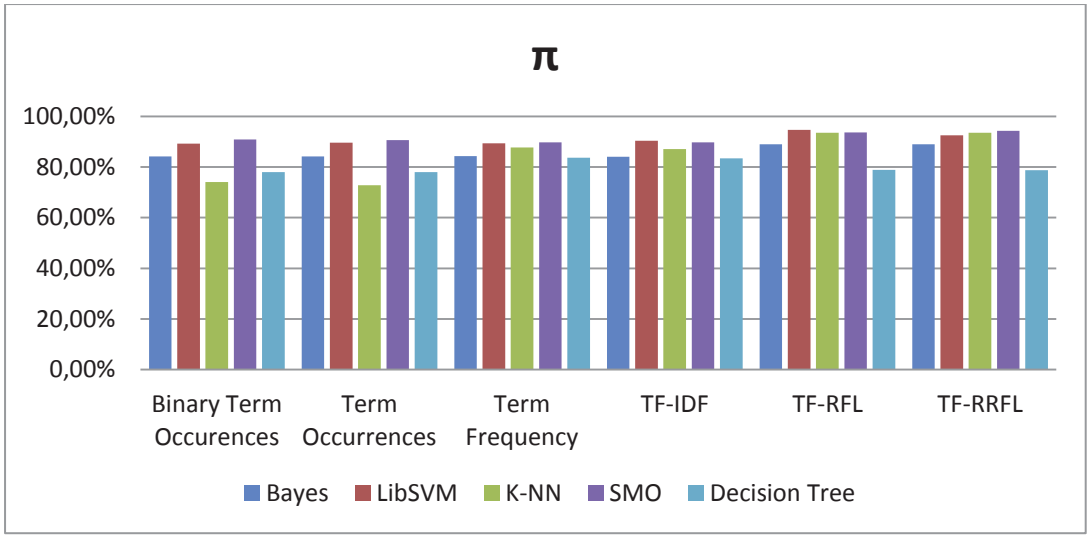

Figura 5-12 Gráfica precisión tercer data set

El grafico anterior representa la precisión de las pruebas aplicadas al data set, en este se puede observar que el mejor resultado lo arroja LibSVM con la representación TF-RFL con una precisión de 94,67% promedio micro. En segundo lugar se encuentra SMO con la representación TF-RRFL la cual da una presión de 94,31% promedio micro y es seguido en tercer lugar por el mismo clasificador pero con la representación TF-RFL con una precisión de 93,64% promedio micro. En promedio entre todos los clasificadores en que arroja la mejor precisión es TF-RRFL con un promedio de 88,32%.

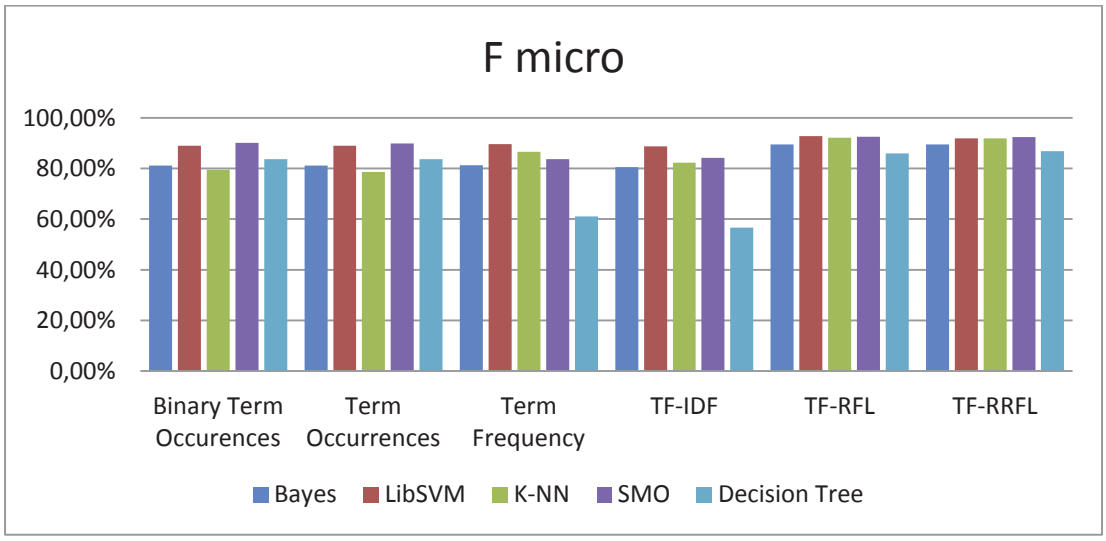

Figura 5-13 Gráfica F micro tercer data set

El grafico anterior representa la métrica F micro de las pruebas aplicadas al data set, en este se puede observar que el mejor resultado lo arroja LibSVM con la representación TF-RFL con un F micro de 92,78%. En segundo lugar se encuentra SMO con la representación TF-

RFL la cual da un F micro de 92,55% y es seguido en tercer lugar por el mismo clasificador pero con la representación TF-RRFL con un F micro de 92,40%. En promedio entre todos los clasificadores en que arroja la mejor F micro es TF-RFL con un promedio de 89,16%.

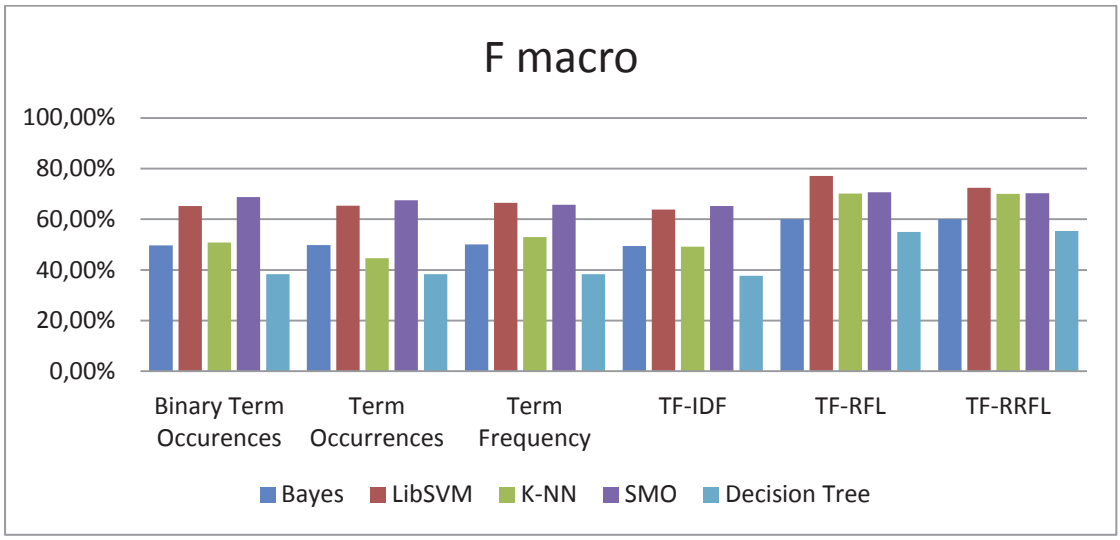

Figura 5-14 Gráfica F macro tercer data set

El grafico anterior representa la métrica F macro de las pruebas aplicadas al data set, en este se puede observar que el mejor resultado lo arroja LibSVM con la representación TF-RFL con un F macro de 77,08%. En segundo lugar se encuentra el mismo clasificador pero con la representación TF-RRFL la cual da un F macro de 72,46% y es seguido en tercer lugar por el clasificador SMO con la representación TF-RFL con un F macro de 70,61%. En promedio entre todos los clasificadores en que arroja la mejor F macro es TF-RFL con un promedio de 81,11%.

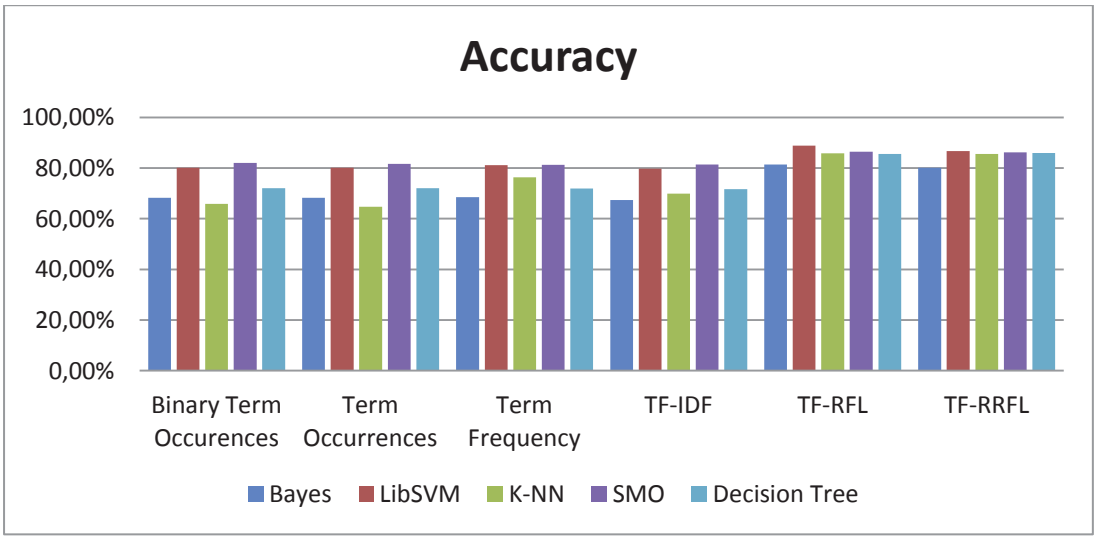

Figura 5-15 Gráfica Accuracy tercer data set

El grafico anterior representa el acierto de las pruebas aplicadas al data set, en este se puede observar que el mejor resultado lo arroja LibSVM con la representación TF-RFL con una acierto de 88,82%. En segundo lugar se encuentra el mismo clasificador pero con la representación TF-RRFL la cual da una acierto de 86,64% y es seguido de muy cerca por el clasificador SMO con la representación TF-RFL con una acierto de 86,50%. En promedio entre todos los clasificadores en que arroja el mejor acierto es TF-RFL con un promedio de 80,57%.

5.5 Análisis por Volumen de Datos

Los gráficos están divididos en las métricas anteriormente descritas, en la línea horizontal se presentan la cantidad de datos aplicados, y la línea vertical representa el porcentaje de la métrica asociada, los colores de las líneas están asociadas las representaciones aplicados en esta investigación.

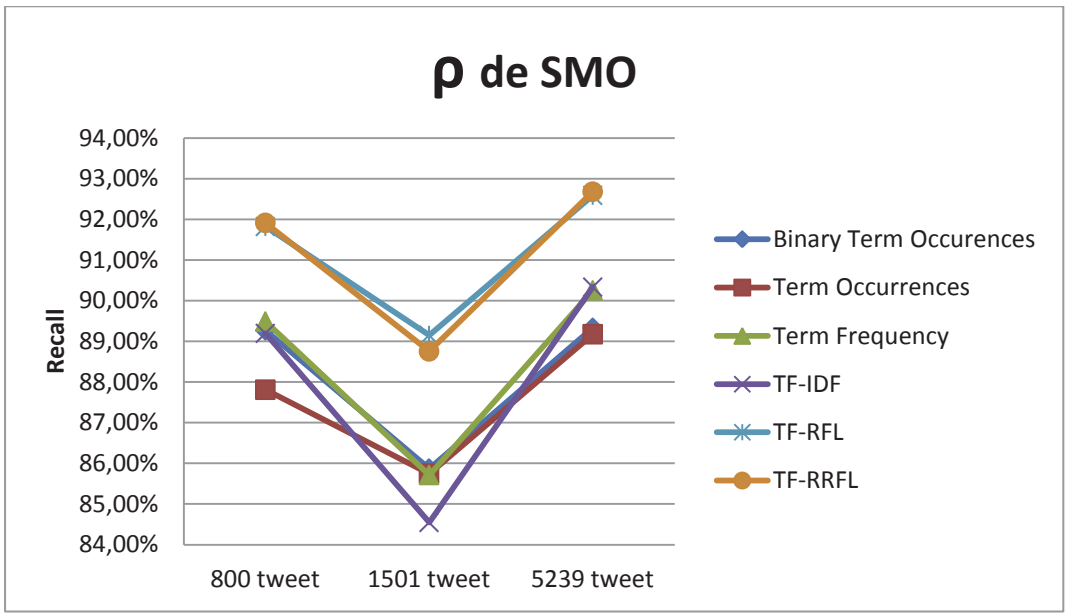

Figura 5-16 Gráfico volumen recall SMO

Como se había analizado en los gráficos de la sección 4.4, los mejores resultados lo arrojan las representaciones TF-RFL y TF-RRFL. Se puede observar que el recall obtiene buenos resultados con un volumen bajos de datos aunque mejora al aumentarlo considerablemente la muestra, pero se logra ver que con un volumen 1501 baja el recall considerablemente.

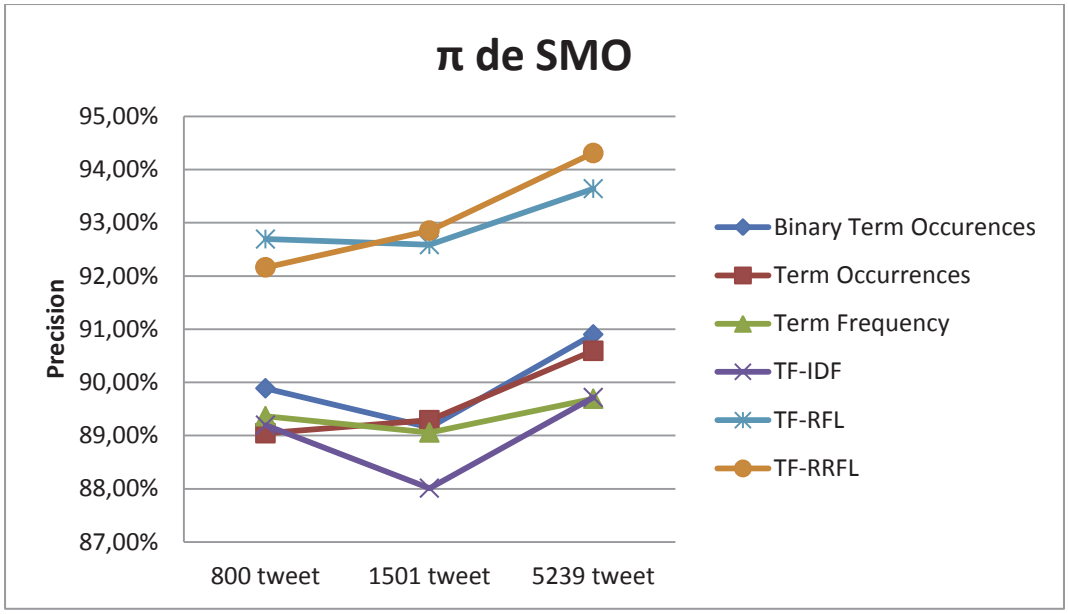

Figura 5-17 Gráfico volumen precisión SMO

Como se había analizado en los gráficos de la sección 4.4, los mejores resultados lo arrojan las representaciones TF-RFL y TF-RRFL. Se puede observar que la precisión obtiene buenos resultados con un volumen bajos de datos aunque mejora al aumentarlo considerablemente la muestra, pero se logra ver que con las otras representaciones con la cantidad de 1501 tweet la precisión baja, pero al aumentar los datos esta vuelve a subir.

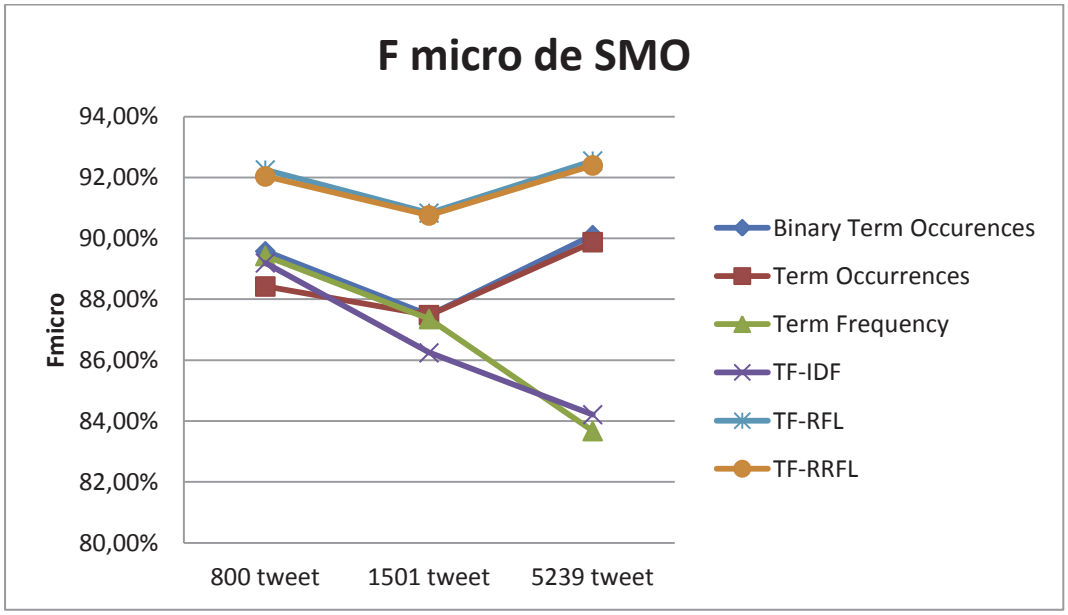

Figura 5-18 Gráfico volumen F micro SMO

Como se había analizado en los gráficos de la sección 4.4, los mejores resultados lo arrojan las representaciones TF-RFL y TF-RRFL. Se puede observar que el f micro obtiene buenos resultados con un volumen bajos de datos aunque mejora al aumentarlo considerablemente la muestra, pero se logra ver que con un volumen 1501 baja el recall considerablemente. La representación por frecuencia y TF-IDF, tienen a bajar su f micro con el aumento de los datos.

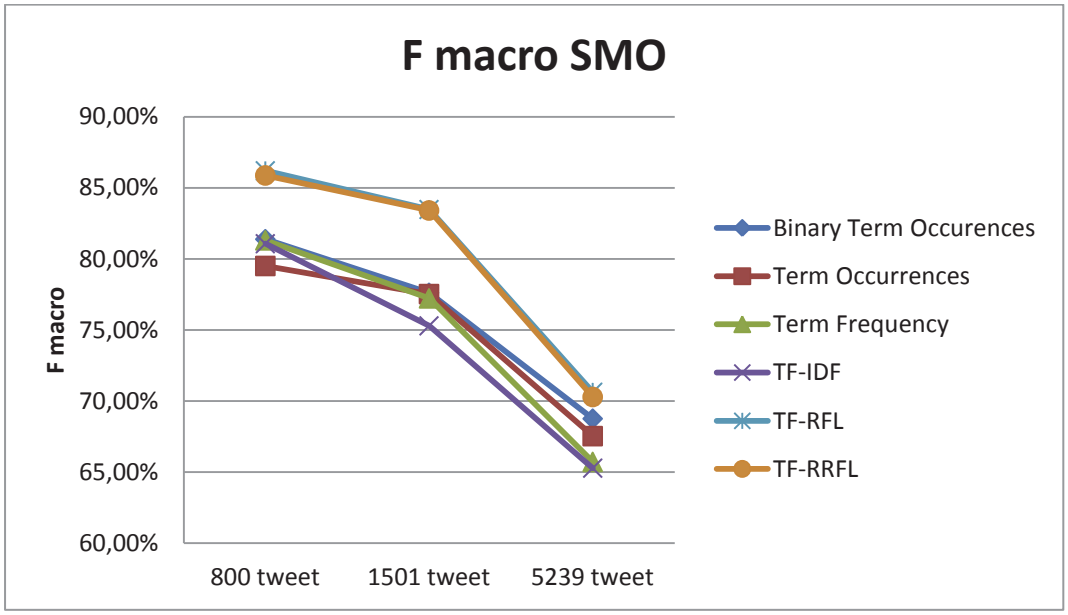

Figura 5-19 Gráfico volumen F macro SMO

Como se había analizado en los gráficos de la sección 4.4 y en este grafico, los mejores resultados lo arrojan las representaciones TF-RFL y TF-RRFL. Se puede observar que el F macro obtiene buenos resultados con un volumen bajos de datos ya que si se empieza aumentar la cantidad de datos el F macro baja, con todas las representaciones.

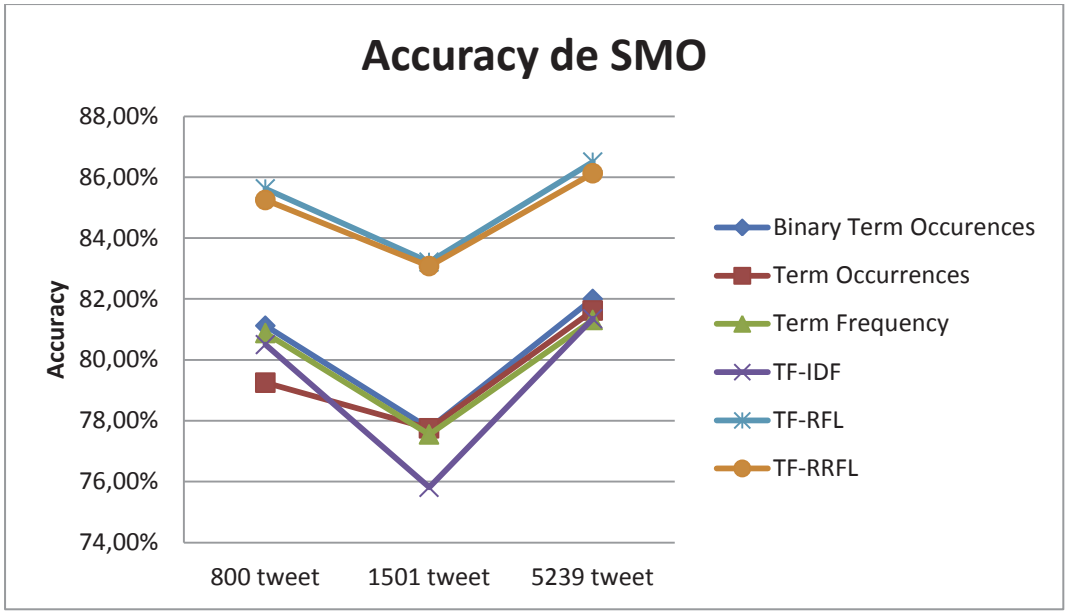

Figura 5-20 Gráfico volumen Accuray SMO

Como se había analizado en los gráficos de la sección 4.4, los mejores resultados lo arrojan las representaciones TF-RFL y TF-RRFL. Se puede observar que el accuracy obtiene buenos resultados con un volumen bajos de datos aunque mejora al aumentarlo considerablemente, pero se logra ver que con un volumen 1501 baja el accuracy considerablemente.

5.6 Conclusiones a las Pruebas Realizadas

Gracias a las pruebas realizadas, se puede concluir que la mejor forma de representar textos cortos para su posterior clasificación automática, entre las representaciones estudiadas en esta investigación es TF-RFL y es seguida de muy cerca TF-RRFL. Esto se pudo concluir gracias a la data de 5239 tweets, entregados por el profesor guía, estos tweet son separados en grupos según su intención, como se describió en la sección 4.3.

Considerando los clasificadores utilizados en esta investigación, se puede concluir que el que entrega mejores resultados, cualquiera sea la representación del textos es SMO con SVM, ya que tiene un buen desempeño, y es seguido de cerca por LIBSVM, y el árbol de decisión, pero este ultimo para entregar buenos resultados, sin embargo, tiene un tiempo muy alto para lograr sus calcificaciones y fue descartado. Por estas razones se propuso realizar un análisis al volumen de la muestra con el clasificador SMO en la sección anterior.

Entre las 3 pruebas realizadas, estas se diferencias entre la cantidad de datos utilizados, por estas pruebas se puede realiza un análisis sobre el tamaño de la muestra, lo que se puede concluir a los gráficos de la sección 4.5 es que al aumentar la muestra a el doble, todas las métricas utilizadas bajan considerablemente, pero al aumentarla a 5239, se puede observar que la mayoría de las métricas aumentan más que con el volumen de 800 datos, pero con

excepción del F macro, la cual baja considerablemente. Como conclusión general se puede tomar que es mejor trabajar con un volumen de 800 datos, 100 por cada clasificación, ya que el tiempo que demora en tratar ese volumen de datos es mucho menor que el de 5239, y los resultados son muy parecidos.

6. Conclusión y Trabajo Futuro

Con la investigación realizada en esta tesis se pudo describir la problemática de la clasificación de textos cortos, o mejor conocidos como mensajes de Twitter, clasificados de acuerdo a la intención del usuario para realizarlos. Gracias a esta investigación se pudo conocer las distintas formas de clasificar y representar un texto, para lograr que el computador pueda imitar el aprendizaje de los humanos. Se pudo plantear la problemática y se explico paso a paso el pre-procesamiento de los datos para poder desarrollar una propuesta que permite dar solución el problema planteado, ocupando el data set para realizar pruebas.

En esta investigación se toma en cuenta las representación de textos en forma vectorial con las representaciones más comunes de esta forma, pero también se toman en cuenta 2 nuevas representaciones mutilase TF-RFL y TF-RRFL los cuales son plateados por 2 profesores de la universidad [11].

Se comprendió que funcionamiento de los métodos de aprendizaje automático depende en gran medida de la elección de la representación de datos en el que se aplican. Por esa razón, gran parte del esfuerzo real en el despliegue de algoritmos de aprendizaje automático entra en el diseño de reprocesamiento de datos y las transformaciones que resultan en una representación de los datos que pueden apoyar la máquina de aprendizaje eficaz. Además se vio reflejado que los algoritmos de aprendizaje actuales tienen incapacidad para extraer y organizar la información discriminante de los datos.

Se logó analizar 6 formas de representación vectorial, con 5 algoritmos de clasificación automática, los cuales dan una clara inclinación de cual representación es mejor, ya cualquiera sea el volumen de datos las mejores resultados los entregan la representación TF-RFL y seguida de cerca por TF-RRFL ya que estas superan en gran medida a las representaciones más conocidas, esto se concluye que se logra gracias a que se le asigna el valor según la relevancia de la palabra en la clasificación, debido a que los otro métodos se basan en la frecuencia dentro de una palabra o la importancia con respecto al largo del texto lo cual puede ser favorables para textos largos pero en el caso de los tweet no. Esta dos representaciones ven la importancia pero dentro de la etiqueta definida logrando los mejores resultados en las métricas utilizadas.

Además se puede concluir del análisis del volumen de la data, es que hay que lograr un equilibro entre la cantidad de datos con respecto al tiempo de ejecución y de los resultados, porque se pudo observar que al aumentar la data al doble de los datos, estos tendían a dar peores resultados, pero al aumentarlos significativamente dan mejores resultados en las métricas aplicadas, pero se invierte un tiempo mayor en logar la clasificación, así como un gasto significativo en memoria del computador no superando en gran medida a los datos de una muestra más pequeña, por lo cual se recomienda sacar una muestra significativa y no muy

grande para el análisis de los datos, para de esta manera poder obtener buenos resultados en poco tiempo de ejecución y sin un gasto mayor de memoria del computador.

Como trabajo futuro queda en el tintero poder abordar las formas de representación de textos las cuales no brindan fácilmente las herramientas de aprendizaje automático así como las de minería de datos, poder representar y trabajar con n-gramas, ya que por tiempo no se pudo realizar la comparación con esta representación. Uno de los desafíos grandes en la clasificación automática de textos cortos o tweet, es poder generar herramientas de representación de textos de manera de grafos, ya que es un amplio campo y según la literatura puede generar mejores resultados que la representación vectorial.

El tipo más sofisticado de procesamiento de texto, consideraremos brevemente, es la etapa de producir un completo análisis sintáctico de una oración. Por esto, queremos decir que cada palabra en una frase está conectado a una estructura única, por lo general un árbol, pero a veces un gráfico a cíclico dirigido. En el análisis, se encuentra la relación de cada palabra en una oración para todos los demás, y por lo general también su función en la frase (por ejemplo, sujeto, objeto, etc.), son muy diferentes los tipos de forma para analizar cada frase asociado a una teoría lingüística de la lengua. Esto no es el lugar para discutir estas varias teorías. Para propósitos de clasificación, podemos restringir a la denominada " análisis libre de contexto". Se puede imaginar que este tipo de árbol de nodos en los que los nodos hoja son las palabras de una frase, las frases en las que se agrupan las palabras son nodos internos, y hay un nodo principal en la raíz del árbol, que por lo general tiene la etiqueta S. Hay un número de algoritmos para la producción de un árbol tal de las palabras de una frase. Muchas investigaciones se han realizado en la construcción de programas de análisis de un análisis estadístico de los bancos de árboles de frases analizado a mano, él más conocido y más utilizado es árbol de sentencias analizadas por el Wall Street Journal y está disponible en los países menos adelantados [7].

Es por esto que este campo puede tener grandes mejoras y es necesario indagar mucho más al respecto, la investigación de la clasificación automática de texto es un gran avance para realizar análisis de redes sociales, de manera de poder obtener la información en el momento que esta tiene la emoción, de manera de mejorar las actuales encuestas que, según el estado de ánimo en el que se realizó a esta la encuesta, puede dejar una apreciación positiva o negativa, así las redes sociales nos plasman la emoción en el momento que esta tiene la emoción correcta al objeto que queremos analizar.

7. Referencias

[1] Coyotl M. "Clasificación Automática de Textos considerando el estilo de Redacción" INADE 2007. Disponible en: http://ccc.inaoep.mx/~villasen/index_archivos/tesis/TesisMaestria-RosaCoyotl.pdf

[2] Joachims T. "Text Categorization with Support Vector Machines: Learning with many relevant features". Proceedings of {ECML}-98, 10th European Conference on Machine Learning. Chemnitz, Germany, Issue 1398., pp. 137-142, 1998.

[3] Fabrizio Sebastiani "Machine Learning in Automated Text Categorization" Consiglio Nazionale Delle Ricerche, Italy. ACM Computing Surveys (CSUR), Volume 34 Issue 1, March 2002, Pages 1-47.

[4] Chih-Chung Chang and Chih-Jen Lin "LIBSVM: A Library for Support Vector Machines", Department of Computer Science National Taiwan University, Taipei, Taiwan. Disponible en: http://www.csie.ntu.edu.tw/~cjlin/papers/libsvm.pdf

[5] Hernández J., Ramírez J. & Ferri C. "Introducción a la minería de Datos". Prentice Hall, Pearson Education, S.A., Madrid, 2004.

[6] Ass K. & Eikvil L. "Text categorization: A survey". Technical Report. Norwegian Computing Center, 1999. Disponible en: http://www.oocities.org/rr_andres/docs/aas99text.pdf

[7] Sholom M. Weiss "Text Mining predictive methods for analiyzing unstructured information", Springer: 9, November 2004, Pages 15-46.

[8] Rapid-I Report the Future. Disponible en: http://rapid-i.com/content/view/181/190/lang,en/

[9] Martis Mauricio, Tesis para optar al grado de Magíster en Ingeniería Informática, titulada "Clasificación Automática de la Intención del Usuario en Mensajes de Twitter", Pontificia Universidad Católica de Valparaíso, Enero 2012.

[10] Man Lan, Chew Lim Tan, Jian Su, and Yue Lu, *Supervise and traditional term weighting methods for automatic text categorization,* IEEE Transactions on Pattern Analysis and Mechine Intelligence 31 (2009), 721-735.

[11] Rodrigo Alfaro and Héctor Allende, "Text Representation in Multi-Label Classification: Two Nex Input Reresentations", PUCV, UTFSM and UAI from Chile. Disponible en: http://www.rodrigoalfaro.cl/documentos/Articulo_ICANNGA-2011.pdf

Anexos

A: Tablas Detalladas de Valores de los Gráficos Analizados

B: Gráficos de Volumen de los Clasificadores

C: Matriz de confusión de las Pruebas