Beneméritа Universidad Autónoma de Puebla

Tésis de Maestría

Combinación de clasificadores para el Análisis de Sentimientos

Author:
Monserrat Ramírez García

Asesor:
Dra. Maya Carrilo Ruíz

Facultad de Ciencias de la Computación

February 2015

Contents

List of Figures

List of Tables

Abbreviations

LAH **L**ist **A**bbreviations **H**ere

Symbols

a	distance	m
P	power	W (Js^{-1})
ω	angular frequency	rads^{-1}

Chapter 1

Introducción

Hoy en día es muy común encontrar en redes sociales, blogs, microblogs, páginas web, entre otras, información u opiniones de los usuarios que expresan su punto de vista en internet acerca de algo. Dicho fenómeno ha generado interés por el análisis de sentimientos (AS), una área del procesamiento de lenguaje natural (NLP) que se encarga de identificar opiniones relacionadas con un objeto. El interés proviene, de que un factor determinante para la toma de decisión de las personas es precisamente la opinión, por ejemplo cuando compramos algún producto en internet, queremos conocer la opinión de los demás acerca del producto que deseamos adquirir. De la misma manera, las empresas tienen como objetivo encontrar indicadores que contribuyan a cubrir las necesidades de sus clientes, mejorando sus productos y servicios, lanzando al mercado productos que con base en las opiniones adquiridas prefieran o soliciten sus clientes y estar al pendiente de su posición con respecto a la competencia. En el ambiente político es importante Conocer la opinión de las personalidades públicas, elegir la propaganda idónea según las preferencias u opiniones de la gente o simplemente elegir el producto mejor valorado por los usuarios [1].

La utilidad de evaluar la opinión pública usando análisis de sentimiento sobre opiniones digitales permite la sustitución de los medios tradicionales como las encuestas y estudios de campo.

Dada la importancia del análisis de sentimientos establecida en los párrafos anteriores en esta investigación se explora la Combinación de Clasificadores para realizar Análisis de Sentimientos.

1.1 Objetivos

1.1.1 Objetivo General

- Mejorar la precision del Analisis de Sentimientos, deniendo una arquitectura que combine varios clasicadores base, buscando mejorar la precision obtenida por cada clasicador.

1.1.2 Objetivos Particulares

- Analizar diferentes Metodos de Analisis de Sentimientos.
- Probar con clasicadores base y determinar cuales se comportan de mejor manera para el problema planteado.
- Identicar arquitecturas que permitan combinar clasicadores que sean aplicables al analisis de sentimientos.

1.2 Estructura de la tesis

La estructura del documento se presenta de la siguiente manera: en el capítulo 1 se enuncian algunos trabajos relacionados con la investigación, en el capítulo 2 se presenta la tarea de análisis de sentimientos, el capítulo 3 presenta la definición de conceptos empleados durante este trabajo de investigación. El capítulo 4 describe lo que es el aprendizaje supervisado, combinación de clasificadores, clasficadores base utilizados y métricas de clasificación. En el capítulo 5 se presentan los experimentos realizados y los resultados obtenidos. Finalmente en el capítulo 6 se presentan las conclusiones obtenidas y por último la bibliografía utilizada en el capítulo 7.

Chapter 2

Análisis de Sentimientos

2.1 Trabajos Relacionados

En esta sección se presentan trabajos de dos áreas importantes para la presente investigación: Análisis de Sentimientos y Combinación de Clasificadores.

Como primera parte se citan los trabajos reportados en el área de AS (Liu, Sentiment Analysis and Opinion Mining, 2012) [2], (Khan, 2009), [3], (Liu, Web Data Mining: Exploring Hyperlinks, Contents and Usage Data, 2011) [4], etc.

Los primeros trabajos en el área fueron desarrollados a partir del año 2000. Se trabajó desde el inicio con enfoques de diccionarios: estadísticos y semánticos. El enfoque basado en aprendizaje automático, fue una herramienta utilizada en las investigaciones de ese tiempo, la popularidad actual de este enfoque para la minería de opinión se origina en el trabajo publicado en el 2002 "Thumbs up?" por Pang y Lee donde utilizan tres métodos de clasificación: Naive Bayes(NB), Máxima Entropía(ME) y Maquinas de Soporte Vectorial (SVM), los tres algoritmos con base a una elección aleatoria, obtuvieron alrededor del 80% de la precisión, siendo SVM el mejor. Posteriormente Dave junto con su equipo en 2003 basados en el trabajo de Pang y Lee enfatizan la selección de características, y utilizan una técnica de Laplaciano suavizado y con ello se mejora al 87% de precisión (para un dataset Particular), sin embargo SVM siempre dio resultados cercanos a este porcentaje. En 2004 Pang y Lee utilizan identificación de subjetividad como un paso de pre procesamiento con el fin de mejorar la precisión del NB.

El análisis de opinión es un problema similar a la tarea de asignación de calificación considerando valores escalares, “estrellas”. A pesar de que el método de SVM daba buenos resultados en clasificaciones binarias, la nueva aproximación, exigía soluciones más sofisticadas. Pang y Lee haciendo frente a este nuevo problema en 2005, realizan un estudio llamado “Seeing Stars”, que propone utilizar SVM en múltiples clases, es decir una contra todas, y Regresión de Maquinas de Soporte Vectorial (SVR) combinándolo con un etiquetado numérico. Los resultados demostraron que la combinación de las SVM con algún otro método de clasificación no supervisado obtiene una mejor precisión. En un trabajo posterior realizado en 2006 por Matt Thomas, Bo Pang, and Lillian Lee [5], también son estudiados algunos otros enfoques con SVM que coinciden con los resultados de los trabajos anteriores.

El desempeño de los métodos de aprendizaje automático es altamente dependiente de la calidad y de la cantidad de datos de entrenamiento. En el artículo publicado en 2006 titulado “Seeing Stars When There Are Not Many Stars”, Goldberg y Zhu propusieron una técnica de aprendizaje semi supervisado operando en un gráfico de datos etiquetados y sin etiquetar. Los autores representan documentos con un gráfico, donde los vértices corresponden a los documentos, y los bordes se dibujan entre documentos similares utilizando una medida de la distancia calculada directamente de las características del documento. A pesar de que su enfoque exhibió un mejor rendimiento que RVS, los autores mencionan que es sensible a la elección de la medida de similitud, y no es capaz de beneficiarse de la utilización de los datos etiquetados adicionales.

En 2005 surgió un nuevo problema, la clasificación de sentimiento contextual, ahora se requería de algoritmos no solo operando en el nivel de la oración, sino que también en el análisis del contexto de cada frase. En función de esto, Shimada y Endo en 2008 proponen analizar calificaciones de nivel de producto-características, nombrando su trabajo como ”Seeing Several Stars”, donde demostraron que la SVR (linear Support Vector Regression), a pesar de ser menos preciso que SVM (Support Vector Machine), produce etiquetas de salida que son más cercanos a los reales. Esta evidencia también apoya la afirmación de Pang y Lee que con el uso de una función ”gradual” en RVS ” los objetos similares reciben necesariamente etiquetas similares”.

En 2007 Osherenko y André demuestran que es posible utilizar sólo un pequeño conjunto de las palabras (diccionario) más afectivas como características, casi sin ninguna

degradación en el rendimiento del clasificador. Sin embargo, el uso directo de los valores de sentimiento de tales diccionarios ha mostrado poco o incluso ningún aumento de la precisión. Los estudios por lo general utilizan frecuencias de palabras. Por ejemplo, en 2007 Devitt y Ahmad identifican palabras de soporte de sentimiento en un documento mediante la herramienta SentiWord –Net. Especificando un poco más, la Aproximación de Diccionario, se basa en un diccionario pre construido que contiene polaridades de opinión de las palabras, como: Inquirer2, WordNet-Affect 4 o la SentiWordNet, que es el diccionario más popular hoy en día. Fahrni y Klenner en 2008; Tsytsarau en 2010. Missen y Boughanem en 2009, utilizan las puntuaciones de polaridad directa, proporcionando un valor de sentimiento en una escala continua. La mayoría de los métodos de los diccionarios agregan los valores de polaridad de una sentencia o documento, y calculan el sentimiento resultante usando algoritmos simples basados en reglas como en el trabajo de Zhu en 2009. Herramientas más sofisticadas, como el Analizador de Sentimiento introducido por Yi et al. en 2003 o el enfoque lingüístico en 2009 por Thet et al., extraen con precisión los sentimientos de algunos temas de destino mediante métodos avanzados que aprovechan las características específicas de dominio, así como las estructuras de frases de opinión y etiquetas. Fahrni y Klenner en 2008, proponen para obtener polaridades, utilizar la coocurrencia de los adjetivos en un corpus. En este caso, la capacidad de adaptación se consigue a través de la construcción de un diccionario de corpus-específico. En cuanto al problema de la falta de disponibilidad de algunas palabras, el método de corpus estadístico propone superar los resultados mediante el uso de corpus suficientemente grandes. Podemos identificar la polaridad de una palabra mediante el estudio de la frecuencia con la que esta palabra aparece en un gran corpus dado de textos. Si la palabra se presenta con mayor frecuencia entre los textos positivos o negativos, entonces tiene una polaridad positiva o negativa respectivamente. Igualdad de frecuencias indican palabras neutras. También es interesante mencionar, que en el trabajo de Ku et al.en 2006, 2007 que las aplicaciones que trabajan con la lengua china son capaces de reconocer la polaridad incluso para decir que el mensaje está oculto, gracias al hecho de que los caracteres fonéticos determinan el sentido de la palabra. Turney et al. Proponen obtener la frecuencia de coocurrencia de términos basándose en las estadísticas del buscador web AltaVista. En 2005 extendiéndose sobre este trabajo, Chaovalit y Zhou utilizan el motor de búsqueda de Google para determinar la coocurrencia de las palabras, lo que aumenta la precisión. Read y Carroll en 2009 extendieron aún más este enfoque, empleando espacios semánticos y similitudes

distributivas como métodos alternativos débilmente supervisados. Un estudio detallado sobre la construcción de diccionarios de este tipo fue realizada por Taboada en 2006, quien menciono algunos de los problemas que se producen debido a la indisponibilidad del modificador "cercano o parecido" o la no persistencia de la producción del motor de búsqueda. Ben He et al. (2008), haciendo uso de métodos estadísticos en el cálculo de la polaridad de opinión, propone utilizar un diccionario de opinión junto con métodos de Recuperación de Información (IR) con el objetivo de recuperar las opiniones de blogs.

El enfoque semántico proporciona valores de sentimiento directamente (al igual que el enfoque estadístico), excepto que se basa en principios diferentes para calcular la similitud entre las palabras. El principio fundamental de todos los enfoques en esta categoría es que semánticamente palabras parecidas deben recibir valores de sentimiento similar. WordNet proporciona diferentes tipos de relaciones semánticas entre las palabras. La posibilidad de eliminar la ambigüedad de sentidos de palabras usando WordNet puede servir como una manera de incluir el contexto de estas palabras en la tarea de análisis de la opinión. Kamps et al. Propusieron utilizar la distancia relativa de la ruta más corta de la relación "sinónimo", con lo que se obtiene una precisión de (70%), con un diccionario dado. Otra forma popular de utilizar WordNet es obtener una lista de palabras de sentimiento expandiendo de forma iterativa el conjunto inicial de sinónimos y antónimos. La polaridad de sentimiento de una palabra desconocida, se determina por el recuento relativo de sus sinónimos: positivos y negativos. De lo contrario, las palabras desconocidas, pueden ser descartadas. Sin embargo es importante tener cuidado con la diferencia entre el sinónimo y la palabra original, como es señalado por Godbole et al. en 2007, sólo debemos tener en cuenta los caminos que van a través de las palabras de la misma polaridad inicial. Con el crecimiento de las redes sociales, el análisis de opinión se ha extendido a los microblogs. Existen aplicaciones, como el análisis de mensajes en microblogs en Twitter, los cuales son capaces de adaptar el modelo utilizado a la evolución de los datos durante el análisis. Recientemente, Tumasjan et al. en 2010 demostró que los sentimientos de mensajes de Twitter se correlacionan con las preferencias políticas, e incluso Bollen et al. en 2010 también demostraron mejorar la predicción del mercado de valores, mediante el análisis de microblogs. Trabajos recientes han identificado varias diferencias entre la minería de opinión en microblogs en comparación con los análisis de la opinión convencional de documentos. La principal diferencia es la disponibilidad de sentimiento o estado de ánimo en los mensajes en microblogs, que

proporciona una buena fuente de datos de entrenamiento para los clasificadores. Pak y Paroubek (2010) realizaron un análisis estadístico de las características lingüísticas de los mensajes de Twitter, en su trabajo reportan patrones interesantes para la clasificación de AS, así también demuestran con un clasificador NB, considerando palabras negadas y con características representadas en n-gramas principalmente bigramas, se logra una buena precisión (aunque, a expensas de bajo recuerdo), se plantea que ésta contribución puede ser útil para aplicaciones de recuperación de información. Bermingham y Smeaton (2010) compararon el desempeño de clasificadores SVM y NB multinomiales (NBM) en datos de microblog y demostraron que en la mayoría de los casos, estos clasificadores dan mejores resultados en opiniones de longitud corta, los mensajes de microblogs son ricos en opiniones. Bifet y Frank en 2010 estudiaron el problema de usar un clasificador adaptable a los datos de correo electrónico. Propusieron utilizar el método de descenso de gradiente estocástico (DGS) con el cuál obtuvieron una precisión más pequeña, pero comparable a la de NBM (67,41% frente a 73,81%).

Como segunda parte del estado del arte, se revisará el área de combinación de clasificadores, citando de manera breve aportaciones importantes en dicha área (Rokach, 2005) , (Kuncheva, 2004), (Jurek, 2013). Para su estudio en una primera etapa se construye un ensamble de clasificadores lo cual involucra un proceso de selección de un conjunto de diferentes clasificadores base. Para este proceso se presentan dos enfoques:

- Utilizar un sólo algoritmo de aprendizaje y diferentes conjuntos de entrenamiento, en donde el objetivo principal es la conversión del conjunto de datos original para obtener una colección de diferentes conjuntos de datos de entrenamiento. Algunas técnicas dividen el conjunto de datos de forma aleatoria, o con la manipulación de la distribución de datos.

- Cada clasificador base es entrenado con el mismo conjunto de datos y utilizando diferente algoritmo de aprendizaje, como resultado se obtienen diferentes modelos de clasificación.

A continuación se presentan algunos trabajos correspondientes al primer enfoque: En 1996 Breiman propone el método de Bagging. Bagging es la técnica más popular para obtener diferentes conjuntos de datos de entrenamiento. Bagging se relaciona con el

enfoque donde el conjunto de entrenamiento es elegido aleatoriamente k veces con reemplazamiento (técnicas bootstrap) de un conjunto de datos original. La ventaja principal de esta técnica es la independencia de los miembros del ensamble por lo tanto pueden ser entrenados en paralelo, lo cual reduce tiempo. La desventaja es que los subconjuntos de entrenamiento generados de forma aleatoria con reemplazamiento no son totalmente independientes. Skurichina y Duin, en 1998 con su método Nice bagging, modifican el método de Bagging, seleccionando los clasificadores base de mejor rendimiento. El modelo final no mejora el rendimiento de los clasificadores base, pero es más estable. En 1999 el método Wagging es propuesto por Bauer y Kohavi, el cual trabaja con el algoritmo de Bagging modificando la distribución de los datos en entrenamiento agregando el "ruido Gaussiano". Con ello se obtuvo resultados con mayor diversidad. En 2007 se propone el algoritmo de Bagging con SVM por Wang y Lin donde buscan la generación de las mejores clases en cada muestra agregada, con ello se mejora el rendimiento de las SVM simples. En 2005 Zhou y Yu con su algoritmo BagInRand, definen una nueva métrica aleatoria modificando cada iteración, con ello aumenta la diversidad entre los clasificadores y mejora el rendimiento del algoritmo de los K vecinos cercanos (kNN). (FALTA REVISAR COMENTARIO ****) En 2009 Gan y Xiao siguiendo con kNN realizan un muestreo de los datos de entrenamiento con una técnica que mejora la generación de clusters en kNN. Siguiendo el mismo enfoque muchos otros trabajos han sido publicados. A continuación se presentan algunos otros métodos de boosting y propuestas de intentos de mejora. En 1999 Freund y Schapire, proponen AdaBoost el cual entrena cada clasificador base con diferente distribución de datos, los resultados son una mayor diversidad entre los clasificadores base. Dicho método resulto exitoso aplicado en modelos inestables. MultiBoosting fue propuesto por Webb en el 2000, donde se agrega ruido Gaussiano a cada peso, con lo cual se logra una mayor diversidad entre los clasificadores base y supera a los algoritmos de bagging aplicados con árboles de decisión. En el mismo año Domingo y Watanabe proponen MadaBoost en el que se limita el peso de cada instancia a su probabilidad inicial, con ello se reduce el problema de sobreajuste y supera el método AdaBoost. Posteriormente otros métodos utilizando SVM fueron propuestos como AdaBoostSVM que ajusta los parámetros del kernel en cada ciclo para obtener un promedio preciso de clasificadores, con esta aportación se reduce el problema de sobreajuste. En 2007 Vezhnevets & Barinova proponen eliminar muestras confusas, eliminando instancias que no son clasificadas correctamente por un modelo bayesiano perfecto, dicha aportación también disminuye el sobreajuste. En 2010 se propone el

algoritmo Bs-kNN en el cual cada iteración del modelo es construida con diferentes conjuntos de características, con ello se incrementa la diversidad entre los clasificadores base y mejora significativamente el rendimiento del kNN simple. Otro enfoque para generar una colección de clasificadores base con el mismo conjunto de datos de entrenamiento es mediante la aplicación de diferentes subconjuntos de características. A continuación se citan algunos de los métodos basados en este enfoque. En 2001 Breiman presenta "Random forest" generando un número grande de árboles individuales, seleccionando variables aleatorias en cada nodo, lo que aumenta la diversidad y supera al método de bagging decisión trees y boosting. En 2003 Bryll et al. Proponen "Attribute bagging" en el cual en cada iteración es tomado un conjunto de características de forma aleatoria, dicho método logra un mejor rendimiento del clasificador final y es más estable comparado con el algoritmo simple de bagging.

Una vez obtenida la colección de clasificadores base, el segundo paso en el proceso de construcción del ensamble es combinar los resultados obtenidos. Existe una técnica que combina todos los clasificadores individuales considerados. Mientras que otra selecciona un subconjunto óptimo de modelos utilizados. Existen diferentes métodos para realizar dicha tarea, la forma más sencilla es con métodos de ponderación, en donde los votos proporcionados por todos los clasificadores se cuentan y la clase que recibe el mayor número de votos, es seleccionada como decisión final. Los métodos probabilísticos, son muy populares y con un esquema efectivo de combinación basado en el Teorema de Bayes, en donde el objetivo es asignar Z patrones dentro de las w_j clases y maximizar las probabilidades. Otro enfoque es el basado en razonamiento evidencial, donde las salidas de todos los clasificadores base son modeladas como distribuciones de probabilidad para todas las clases consideradas y luego son tratadas como piezas de evidencia, usualmente estas técnicas son aplicadas a clasificadores base entrenados con diferentes métodos de aprendizaje. La combinación de las decisiones de clasificadores individuales es basada en la selección del subconjunto óptimo de clasificadores, lo que resalta, que no todos los modelos generados contribuyen al proceso de toma de decisión, sin embargo solo el grupo seleccionado puede obtener el mejor rendimiento posible. La selección del ensamble puede mejorar el rendimiento final en términos de precisión y eficiencia. Dicho enfoque permite optimizar los costos computacionales, reduciendo el número de clasificadores base, pues se ha probado que se pueden obtener mejores resultados en comparación con el conjunto original. Las técnicas de selección son divididas en dos

categorías: selección estática y selección dinámica. En la primera técnica se nomina un subconjunto de modelos que son seleccionados una sola vez al inicio y es fijo para todas las muestras de prueba. En la selección dinámica dicho proceso se realiza para cada instancia nueva individualmente, en función de sus características. El problema clave para ambas técnicas es el criterio de selección. El criterio más popular es la diversidad de medidas y el rendimiento individual y combinado.

A continuación se presentan algunos métodos estáticos para la selección de clasificadores base. En 2005 se propone aplicar Algoritmos Genéticos GA como herramienta de optimización, utilizando un criterio de selección con base al rendimiento del conjunto final y a la diversidad entre los clasificadores. "Greedy approach" propuesto en 2008 por Abdelazeem, en el cual se seleccionan los N mejores clasificadores, eliminando o añadiendo modelos específicos para maximizar la mejora del rendimiento, el criterio de selección se basa en la precisión de los clasificadores individuales. En el mismo año Shi y Lv, proponen ASDM que se enfocan en una aplicación de selección de atributos para obtener diversos clasificadores base, donde el criterio de selección es la diversidad entre los clasificadores base y el rendimiento del conjunto final. Zhiqiang y Balaji en 2007 presentan el método EMO y ESW, el primero se enfoca en una aplicación de programación lineal para encontrar los modelos más eficientes seleccionando los clasificadores individuales con mejor precisión, el segundo también se enfoca en aplicación de programación lineal con la diferencia que es para calcular los pesos de los clasificadores base. En 2011 Diao y Shen proponen "FRFS" enfocado en una aplicación de selección de características difusas rígidas para seleccionar grupos de clasificadores base siendo el criterio de selección la independencia de los clasificadores. En el mismo año Pillai et al. Proponen HSM que selecciona diferentes subconjuntos de clasificadores para cada clase, y busca el mejor rendimiento del conjunto final. Finalmente se citan algunos de los métodos dinámicos para selección de clasificadores base. En 1997 Woods et al., propone DCS-LA que selecciona los clasificadores con la más alta precisión en la región pequeña cercana a los patrones de prueba, siendo el criterio de selección la mejor precisión local de los clasificadores base. En 2007 Ko et al., propone KNORA que selecciona grupos de clasificadores que clasifican correctamente los K vecinos cercanos de los patrones de prueba, buscando seleccionar también aquella arquitectura con la mejor precisión local de los clasificadores. Xiao and He en 2009 seleccionan un conjunto de clasificadores base con

una función de aptitud que combina la precisión y la diversidad del ensamble, el criterio de selección es la diversidad entre los clasificadores en la región local. Batista et al., en 2011 propone dos métodos OP-ELIMINAT que selecciona métodos de clasificación que clasifican correctamente los k vecinos cercanos de los patrones de prueba y OP-UNION que para cada vecino de los patrones de prueba, selecciona un numero k de clasificadores, que clasifiquen correctamente. El criterio de selección para ambas propuestas es la precisión local de los clasificadores base.

El análisis de sentimientos (AS), también llamado minería de opinión es un campo de estudio que analiza las opiniones, sentimientos, evaluaciones, actitudes de las personas hacia entidades como productos, servicios, organizaciones, individuos, cuestiones, eventos, tópicos y sus atributos. Dicha área tiene un amplio rango de aplicaciones casi en todos los dominios, la industria es la más interesada en la proliferación de aplicaciones comerciales, además sin mencionar que por primera vez en la historia, se cuenta con una gran cantidad de información en los medios de comunicación social y la web en general, lo cual ha originado que el análisis de sentimientos sea el centro de investigación de los medios sociales. La investigación de AS no solo ha tenido un gran impacto en NLP (Natural Lenguage Processing), si no también ha tenido un profundo impacto en las ciencias de la administración: ciencias políticas económicas y sociales.

Las opiniones son fundamentales para casi todas las actividades humanas, porque son importantes factores de influencia en nuestros comportamientos.

Los indicadores más importantes de sentimientos, son las palabras que expresan sentimiento, llamadas palabras de opinión (opinión words). Estas son palabras que comúnmente son usadas para expresar sentimientos positivos o negativos. Por ejemplo, good, wonderful y amazing son palabras que expresan sentimiento positivo, en cambio bad, poor y terrible son palabras que expresan sentimiento negativo, a dichas palabras se les conoce comúnmente como lexicón de opiniones (sentiment lexicon o opinion lexicon). A pesar de que las palabras y frases con sentimiento son muy importantes para el análisis de sentimientos no son suficientes para obtener éxito, la tarea es mucho más compleja, es decir que el lexicón de opiniones es necesario pero no suficiente para el AS. A continuación se describen algunas situaciones que hacen de AS un problema complejo.

- Una palabra que expresa un sentimiento negativo o positivo puede tener orientaciones opuestas, según el contexto de la oración. Por ejemplo "suck" usualmente indica un sentimiento negativo como el la oración "This camera sucks", y en otra oración como "This vacuum cleaner really sucks" tiene un sentimiento positivo.

- Una oración que contiene una palabra considerada como expresión de sentimiento, puede no expresar un sentimiento. Dicho fenómeno ocurre en oraciones interrogativas y condicionales. Por ejemplo, "Can you tell me which Sony camera is good?" y "If I can find a good camera in the shop, I will buy it.", en donde ambas oraciones contiene n la palabra "good" y en ninguno de los dos casos expresan algún sentimiento positivo o negativo. Sin embargo no todas las oraciones interrogativas y condicionales no expresan algún sentimiento, como en la oración "Does anyone know how to repair this terrible printer" y "if you are looking for a good car, get Toyota Camry."

- Oraciones Sarcásticas como "What a great car! It stopped working in two days.", estas oraciones no son muy comunes en opiniones de productos o servicios, pero son muy comunes en discusiones políticas.

- Existen oraciones que no contienen palabras que expresan algún sentimiento, es decir oraciones objetivas que expresan información factual y sin embargo son oraciones con sentimiento negativo o positivo. Un ejemplo es "Esta lavadora utiliza bastante agua" lo cual expresa una opinión negativa acerca de la lavadora. Otra oración es "Después de dormir dos días en el colchón, se le ha formado un valle en medio", que también es una opinión negativa acerca del colchón.

Detección de opiniones Spam

Las opiniones spam han llegado a ser el mayor problema, ya que cualquier persona tiene acceso a la web y es libre de expresar una opinión sin tener la necesidad de identificarse, lo que ha originado consecuencias indeseables, puesto que personas con identidades ocultas e intenciones maliciosas y haciéndose pasar por público en general realicen publicaciones falsas para promover o bien desacreditar algún producto, servicio, organización o individuos sin ser descubiertas sus verdaderas intenciones. Dichos individuos son llamados escritores de falsas opiniones (opinión spammers) [25].

El problema de Análisis de Sentimientos

Las opiniones y los sentimientos tienen una característica importante a diferencia de la información factual, son subjetivos, por lo que se necesita analizar las opiniones de varias personas y no solo de una. El tamaño influye en la dificultad de las opiniones, pues es un factor determinante para lograr una alta precisión en el análisis de sentimiento. Por ejemplo las opiniones obtenidas de twitter que son de a lo mas 140 caracteres, las hace opiniones más centradas y más enfocadas con poca información irrelevante, siendo más fácil de analizarlas que por ejemplo una opinión en un foro en donde las personas interactúan una con la otra y la dimensión de la opinión dependerá del dominio del tema. Las opiniones sobre productos y servicios son generalmente más fáciles de analizar. Discusiones sociales y políticas son mucho más difíciles debido al tema complejo y el sentimiento, expresiones, sarcasmos e ironías.

Chapter 3

Marco Teórico

La clasificación de sentimientos es también comúnmente conocida como *clasificación de sentimiento a nivel documento*, ya que se considera todo el documento como una unidad. El problema se define de la siguiente manera:

Dado un documento de opinión de evaluación d , determinar toda la polaridad o sentimiento que el autor de la opinión expresa acerca de alguna entidad. Es decir determinar las tuplas existentes de acuerdo a la ecuación (num).

Para este problema existen dos formulaciones basadas en los tipos de valores que toma s, si s toma valores categóricos como positivo, negativo, entonces es un problema de clasificación. Si s toma valores numéricos o puntuaciones dentro de un rango, entonces es un problema de regresión.

Primero se discute el problema de clasificación para predecir la categoría de la clase. Las técnicas más utilizadas para clasificación de documentos utilizan aprendizaje supervisado, aunque también hay métodos no supervisados, la diferencia de ambos se explican gráficamente en la figura 1.

Para la presente investigación se utilizó aprendizaje supervisado, que se describe en la siguiente la sección.

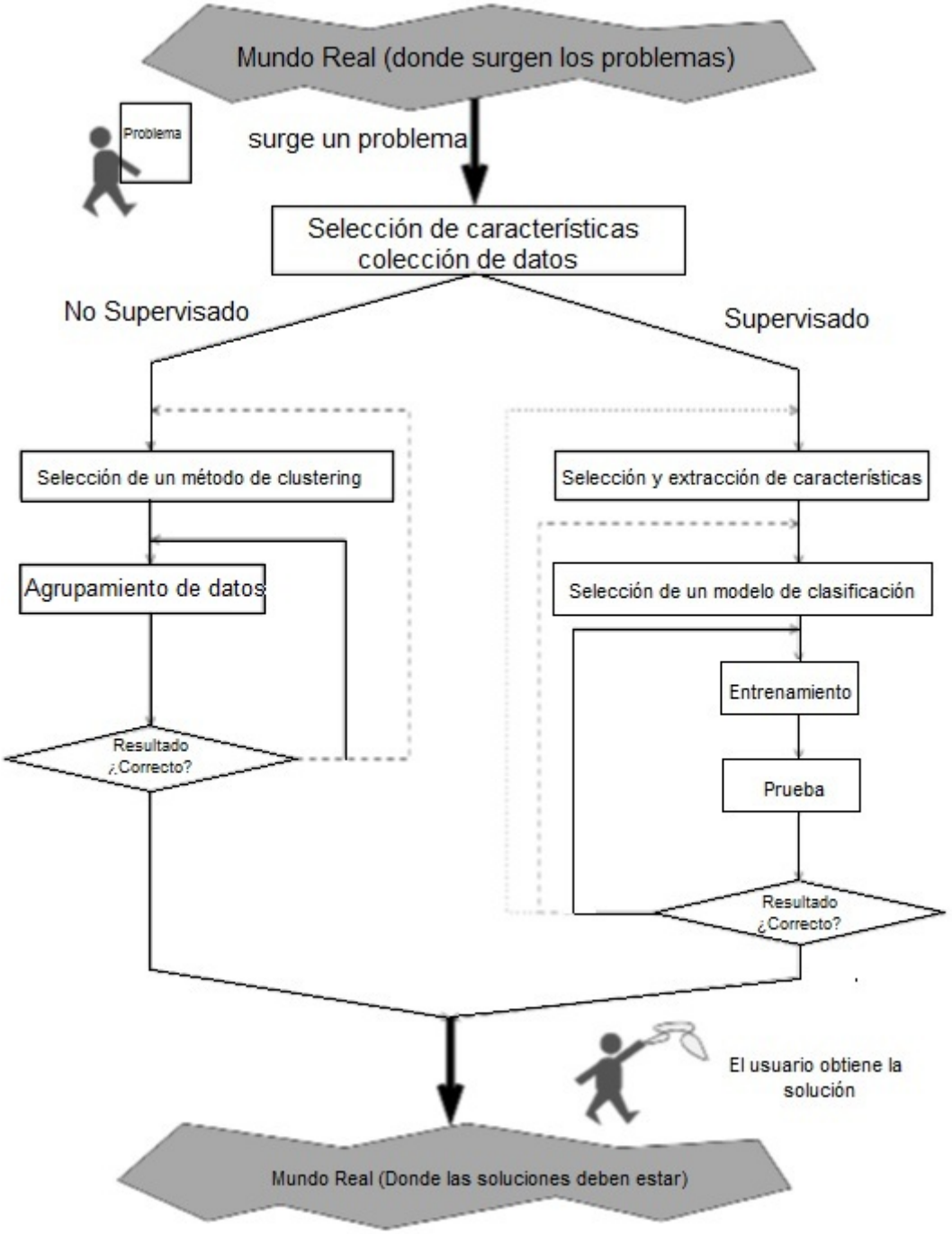

FIGURE 3.1: Aprendizaje Supervizado, y no supervizado

3.1 Opinión

Para definir el concepto de opinión, se presenta un ejemplo:
Posted by: John Smith Date: September 10, 2011 "(1) I bought a Canon G12 camera six months ago. (2) I simply love it. (3) The picture quality is amazing. (4) The battery life is also long. (5) However, my wife thinks it is too heavy for her."

La pregunta es: ¿Qué extraer de la opinión?

Del ejemplo anterior podemos detectar que contiene frases con un sentimiento positivo y negativo acerca de la cámara G12. La oración (2) expresa una opinión positiva acerca de la cámara, la oración (3) expresa una opinión positiva de la calidad de las imágenes, la oración (4) expresa una opinión positiva de la vida de la batería, y la expresión (5) es una opinión negativa acerca del peso de la cámara.

Una opinión consiste de dos componentes clave: un objeto g y un sentimiento s del objeto. *(g,s)* En donde g puede ser una entidad o aspecto de la entidad acerca de la opinión expresada y s es un sentimiento positivo, negativo, neutro o una puntuación que expresa la fuerza o intensidad del sentimiento. Positivo, negativo y neutro son llamados sentimientos (u opiniones) orientaciones o polaridades. Por ejemplo la oración (3) podría descomponerse de la siguiente manera:

(Cannon-G12, picture-quality)

Otro aspecto importante es que la opinión habla acerca de dos personas, quien son llamadas fuentes de opinión (opinión sources) y emisor de la opinión (opinión holder). Y finalmente también la fecha es importante ya que frecuentemente se quieren conocer las opiniones a lo largo del tiempo y como van cambiando.

3.1.1 Definición de opinión

Una opinión es una cuádrupla

$$(g, s, h, t) \tag{3.1}$$

Donde g es la opinión, s es el sentimiento de la opinión y h es la *(opinión holder)*, y t es el tiempo o la fecha en que se expresa la opinión.

Una entidad e es un producto, servicio, tópico, persona, organización o evento. Se describe con un par e: (T, W), donde T es una jerarquía de partes, subpartes, etc. Y W es un conjunto de atributos de e.

Por ejemplo continuando con el ejemplo de la cámara, un modelo de cámara en particular en una entidad y sus atributos son la calidad de la imagen, el tamaño, peso y su conjunto de partes son el lente, visor y la batería. La batería a su vez también tiene un conjunto de atributos, la vida de la batería y el peso de la misma. De lo anterior, puede concluirse que una entidad se define como una descomposición jerárquica de sus partes, donde la raíz es la entidad. Las jerarquías de dos niveles se pueden simplificar utilizando términos llamados aspectos o características para denotar partes y atributos.

Después de la descomposición antes mencionada se puede redefinir la opinión como:

Una opinión es una quíntupla

$$(e_i, a_{ij}, s_{ijkl}, h_k, t_l) \tag{3.2}$$

Donde e_i es el nombre de la entidad, a_{ij} es un aspecto de e_i, ,s_{ijkl}, es el sentimiento en el aspecto a_{ij} de la entidad e_i, ,h_k es el autor de la opinión, y t_l es el tiempo en el cual es expresada la opinión por h_k. Es decir una opinión está dada por s_{ijkl}, que está dada por un autor h_k acerca de aspectos a_{ij} de una entidad e_i en un tiempo t_l.

Ahora se definirá el concepto de modelo de entidad, un modelo de documentos de opinión y el objetivo de la minería, también nombrados aspectos basados en minería de opinión

3.1.2 Modelo de entidad

Una entidad e_i es representada por un conjunto de aspectos, A_i =$\{a_{i1}$,a_{i2},. . . ,a_{in} $\}$. Cada aspecto a_{ij} ϵ A_i de la entidad puede ser expresada por algún conjunto finito de expresiones de aspectos AE_{ij} = $\{$ ae_{ij1},ae_{ij2},. . . ,$ae_{ijm}\}$.

3.1.3 Modelo de documentos de opinión

Un documento de opinión d, contiene opiniones sobre un conjunto de entidades $\{e_1,e_2,\ldots,e_r\}$ expresadas por un conjunto de autores de opinión $\{h_1,h_2,\ldots,h_p\}$. La opinión de cada entidad e_i expresa la entidad misma y su subconjunto de aspectos.

Ahora bien el objetivo de la minería de opinión es:
Dada una colección de documentos de opinión D, descubrir las opiniones o quíntuplas $(e_i, a_{ij}, s_{ijkl}, h_k, t_l)$ en D.

Para la realización de este objetivo son necesarias las siguientes tareas:

- Extracción de entidades y agrupamiento
- Extracción de aspectos y agrupamiento
- Determinación del autor de la opinión y el tiempo de la opinión.
- Generación de la quíntupla

La dificultad de la minería de opinión radica en que todas las tareas anteriores son problemáticas recientemente abordadas y por tanto no resueltas, además también hay problemáticas en la sintaxis usada puesto que una frase puede no mencionar explícitamente algunas piezas que están implícitas como pronombre, convenciones y contexto.

3.1.4 El Problema de Clasificación

La Clasificación es la tarea de predecir una variable discreta *"y"* usando un conjunto de características x_1, x_2, ..., x_n como variables independientes. Para realizar el entrenamiento del clasificador se necesita una función hipótesis h de una colección de ejemplos de entrenamiento. Dicha colección tiene la forma *(X,Y)* y usualmente se refiere a un conjunto de datos *(dataset)*. Cada entrada del conjunto de datos es una tupla *(x,y)*, donde x es el conjunto de características y y es la clase o etiqueta la cual es una variable discreta con c posibles categorías. Cuando los resultados posibles son restringidos a valores binarios, $y_i \epsilon +1, -1, \forall$ i ϵ $1, ..., N$ [6].

3.2 Análisis de Sentimientos usando aprendizaje no supervisado

Un clasificador no supervisado basado en la opinión es el propuesto por Turney. Dicho clasificador decide el carácter positivo o negativo de un documento en base a la orientación semántica de los términos que aparecen en el mismo, pues son el factor dominante para la clasificación de sentimientos. Los patrones sintácticos están compuestos en base a etiquetas POS. A continuación se presenta el algoritmo de Turney [7] que consta de tres pasos:

Paso 1. Dos palabras consecutivas son extraídas si sus etiquetas POS corresponden a alguno de los patrones de la tabla 3.1. Por ejemplo, el patrón 2 significa que dos palabras consecutivas se extraen si la primera palabra es un adverbio, la segunda palabra es un adjetivo y la tercera palabra no es un sustantivo. Como en el ejemplo *"Este piano produce sonidos hermosos"*, *"sonidos hermosos"* satisface el primer patrón.

Paso 2. Se estima la orientación del sentimiento (SO) de las frases extraídas mediante la información mutua puntual llamada medida PMI *(Point-wise Mutual Information (PMI))*.

$$PMI(term_1, term_2) = log_2\left(\frac{(Pr(term_1 \& term_2))}{Pr(term_1)Pr(term_2)}\right) \tag{3.3}$$

PMI mide el grado de dependencia estadística entre dos términos, $Pr(term_1 \& term_2)$ es la probabilidad de coocurrencia del termino 1 y el termino 2 y $Pr(term_1)$ y $Pr(term_2)$ es la probabilidad de ocurrencia simultanea de los dos términos si son estadísticamente independientes. La orientación de sentimiento *(SO)* de una frase se calcula en función de su asociación con las palabras de referencia positiva y la palabra de referencia negativa.

$$SO(phrase) = PMI(phrase, "excellent") - PMI(phase, "poor") \tag{3.4}$$

Las probabilidades son calculadas mediante la emisión de consultas, en donde para cada búsqueda se suele dar un número de documentos relevantes, el cual es el número de éxitos.

	Primera palabra	Segunda Palabra	Tercera palabra
1	Adjetivos,	Sustantivos	cualquiera
2	Adverbios, Adjetivos	Adjetivos	No sustantivos singulares ni plurales.
3	Adjetivos	Adjetivos	No sustantivos singulares ni plurales.
4	Sustantivos Singular	Adjetivos	No sustantivos singulares ni plurales.
5	Sustantivos Plural	Verbos	Cualquiera

TABLE 3.1: Etiquetas POS

En el trabajo presentado por Turney en 2002, el motor de búsqueda de Alta Vista fue utilizado por que tiene un operador *"NEAR"*, para limitar la búsqueda a documentos que contienen las palabras a menos de diez palabras de uno al otro en cualquier orden.

Dados los éxitos de consulta, el número de aciertos obtenidos se pueden calcular como:

$$SO(phrase) = log_2 \left(\frac{hits(phraseNEAR"excellent")hits("poor")}{hits(phraseNEAR"poor")hits("excellent")} \right) \tag{3.5}$$

Paso 3. Dada una opinión, el algoritmo calcula el promedio *SO* de todas las frases en la opinión, y clasifica las opiniones como positivas si el promedio de *SO* es positivo y negativo en otro caso. Otro enfoque no supervisado, es el método basado en léxico, el cual usa un diccionario de palabras de sentimiento con tamaño y orientaciones asociadas y que incorpora intensificación y negación para calcular la puntuación de sentimiento para cada documento [8].

3.3 Análisis de Sentimientos usando aprendizaje supervisado

El Análisis de sentimientos *(AS)* es usualmente formulado como un problema de clasificación de texto en el cual son consideradas dos clases: positiva y negativa. Por lo general la clase neutra no es utilizada.

Desde que el *AS* se definió como un problema de clasificación, se han aplicado métodos de aprendizaje supervisado como Máquina de Soporte Vectorial *(SVM)*, Naive Bayes *(NB)*, entre otros. El primer trabajo realizado que tomo éste enfoque de clasificación fue realizado en [3], en el cual se clasificaron opiniones de películas considerando dos clases,

usando unigramas (bolsa de palabras) como características y los dos clasificadores antes mencionados.

Para la realización del análisis de sentimiento, primero se necesita representar las opiniones de manera computacional para su análisis, para la construcción de la representación de los datos, es importante considerar que las palabras relacionadas con el tema del documento a analizar son las características principales, y es aquí principalmente donde se presenta la primera dificultad del problema, puesto que la clave para poder obtener resultados buenos en el Análisis de Sentimientos es la ingeniería de selección del conjunto de características efectivas [9].

A continuación se presentan algunas características utilizadas para la representación de documentos en el *AS*.

- **N-gramas**

Es una representación tradicional en recuperación de la información que consiste de palabras individuales (unigramas), o cunjuntos de palabras (n-gramas) con sus frecuencias asociadas. En algunos casos podemos representar mejor un concepto mediante la unión de n palabras que se encuentran adyacentes al término principal, lo que se le conoce como *n-gramas*. La importancia de esta representación radica en que la posición de las palabras son consideradas, puesto que el significado de una palabra, no tiene sentido sin las palabras adyacentes que le acompañan en cualquier texto, por lo que la posición de una palabra afecta potencialmente en el sentido del significado de la oración, es decir el sentimiento o la subjetividad dentro de una unidad textual.

- **Bigramas**

Un bigrama o digrama, es un caso especial del n-grama, es un grupo de dos letras, dos sílabas, o dos palabras. Los bigramas son utilizados comúnmente como base para el simple análisis estadístico de texto. Se utilizan en uno de los más exitosos modelos de lenguaje para el reconocimiento de voz [10].

- **Partes de la oración (POS)**

Una técnica de representación muy utilizada se basa en las reglas lingüísticas, donde las palabras y frases son categorizadas como sustantivos, verbos, adjetivos y adverbios. De acuerdo con Turney, son características gramaticales que tienen la capacidad de expresar subjetividad [7]. Existen investigaciones enfocadas principalmente en adjetivos y adverbios, como en el trabajo reportado por Farah Benamara et al [11], en donde expone que las expresiones de una opinión dependen principalmente de algunas palabras, por ejemplo, la palabra *"bueno"* es utilizada comúnmente para una opinión positiva, y la palabra *"malo"*, para algo negativo, dichas palabras son identificadas como adjetivos en términos lingüísticos.

En general los adjetivos son importantes indicadores en una opinión, son considerados características especiales, sin embargo no significa que otras partes de la oración no contribuyan a la expresión de sentimientos. Existen trabajos en donde los sustantivos, verbos, adverbios y sustantivos subjetivos también han tenido buenos resultados [12].

- **TF-IDF *(term frequency-inverse document frequency)***

Es un esquema de ponderación de términos comúnmente utilizado para representar documentos de texto como vectores, que se ajusta al modelo denominado bolsa de palabras, donde cada documento es representado como serie de palabras sin orden. Se trata de una medida estadística de cuán importante es una palabra para un documento en un corpus. Dicha técnica es utilizada para hacer ranking u ordenaciones de los resultados de búsqueda, generación de resúmenes de texto, agrupación y clasificación de documentos, identificación de la autoría de algún texto, recomendación de documentos, etc.

Cálculo del TF

Un término t_j que aparece muchas veces en un documento d_i es más importante que otro que aparece pocas.

$$tf_{ij} = \frac{(n_{ij})}{\sum_{i=1}^{N} n_{ij}} = \frac{(n_{ij})}{|d_i|} \tag{3.6}$$

Donde n_{ij} es el número de veces que aparece el término t_j en el documento d_i y $\sum_{i=1}^{N}$ es la sumatoria del número de veces que aparece el término t_j en todos los documentos.

Calculo del IDF

Un término t_j que aparece en pocos documentos, discrimina mejor que uno que aparece en muchos.

$$idf_j = log\left(\frac{N}{n_j}\right) \tag{3.7}$$

Donde N es el número total de documentos, y n_j es el número de documentos que contiene el término t_j.

Representación Final del documento

Cada elemento queda representado como un vector de características d_j:

$$d_j = (d_{j1}, ..., d_{jn}) \tag{3.8}$$

$$\text{donde, } d_{ij}{=}tf_{ij}{*}idf_{ij}$$

Es decir finalmente se seleccionan n términos con los valores más altos en todos los documentos.

1. **Teoría de la valoración utilizando reglas sintácticas**

La teoría de la valoración propuesta por Peter R.R White [13], se ocupa de los recursos lingüísticos por medio de los cuales las personas expresan alguna opinión. Particularmente del lenguaje (expresiones lingüísticas), la valoración, la actitud y la emoción del conjunto de recursos que explícitamente posicionan de manera interpersonal las propuestas y proposiciones textuales. Es decir trabaja con los significados de las palabras que hacen variar o modificar los términos del compromiso del hablante en sus emisiones, es decir, que modifican lo que está en juego en la relación interpersonal.

Dicha técnica fue implementada por Víctor Morales en su trabajo [14], que haciendo uso de un diccionario de aptitud, el cual contiene características de la teoría de la valoración *(juicio, apreciación y afecto)*, utilizan sintagmas adverbiales con el fin de que dichas reglas obtengan una valor más preciso acerca del significado de cada palabra. El objetivo es contabilizar los valores de positivo, negativo, juicio, apreciación y afecto, que están presentes en una opinión cualquiera, como si se tratase de una bolsa de palabras

ponderada, sin embargo las reglas sintácticas juegan un papel primordial en este proceso, ya que dependiendo del tipo de regla, los valores pueden aumentar, disminuir, o intercambiarse, afectando de esa manera los valores finales asignados al sentimiento de cada opinión.

Chapter 4

Combinación de Clasificadores

La idea de un ensamble de clasificadores, es combinar un conjunto de clasificadores para resolver una tarea en conjunto, en donde el objetivo principal es combinar las salidas de los clasificadores base, para generar una salida en donde sea considerados todos los clasificadores y dicha salida sea mejor que la obtenida por cualquier clasificador base.

Un ensamble de clasificadores es un grupo de clasificadores quienes individualmente toman decisiones que son fusionadas de alguna manera para finalmente obtener una decisión por consenso.

La selección de los clasificadores base se puede realizar de dos maneras: estático o dinámico. En el enfoque estático, se aplica el mismo subconjunto de clasificadores base que se seleccionan para todas las muestras de prueba, en el enfoque dinámico la selección se realiza para cada nueva instancia individualmente.

Posterior a la obtención de la colección de los clasificadores base, el siguiente paso es combinar las salidas en orden de obtener una decisión final, en esta fase, las principales cuestiones que se deben considerar están relacionados con el tipo de información que se va a combinar y qué método de combinación se va a aplicar.

Otro aspecto importante son las salidas de los clasificadores, diferentes métodos de combinación utilizan diferentes tipos de salidas de clasificadores base, por ejemplo una etiqueta de clase o una distribución de probabilidad pueden ser utilizados. Un enfoque alternativo es utilizar predicciones como un conjunto de atributos para formar una función de combinación en términos de meta-aprendizaje [15]. En años pasados,

estudios experimentales realizados por la comunidad de machine learning, mostraron que la combinación de las salidas de múltiples clasificadores reducen la generalización del error. Los métodos de ensamble son muy efectivos, debido principalmente a que varios tipos de clasificadores tienen sesgos inductivos, y provocan que la diversidad de los clasificadores utilizados reduzca el error de la varianza, sin incrementar el error bias [16].

La combinación de clasificadores y por lo tanto la creación de ensamble de clasificadores fue propuesto para mejorar los resultados obtenidos por los clasificadores base. La llave para producir un ensamble exitoso, es elegir los métodos de clasificación apropiados y seleccionar los clasificadores base indicados para el problema planteado.

4.1 Definiciones importantes

Un clasificador es una función

$$D : R_n \rightarrow \Omega \tag{4.1}$$

En el "modelo canónico de un clasificador" [17], se consideran un conjunto de c funciones discriminantes $G = g_1(x), ..., g_c(x)$,

$$g_i : R^n \rightarrow R \tag{4.2}$$

$$i=1,...,c$$

Cada uno produciendo un puntaje para la clase respectiva. Por lo general, x está etiquetado en la clase con la puntuación más alta. Esta elección de etiquetado se denomina la *regla de máxima afiliación*,la cual se describe a continuación:

$$D(x) = w_{i*} \epsilon \Omega \leftrightarrow g_{i*}(x) = min_{i=1...c}\{g_i(x)\} \tag{4.3}$$

Donde w_i son las caracteristicas de cada instancia, Ω es el dominio del conjunto de caracteristicas. g_i son las funciones discriminantes de cada clase o etiqueta. La ecuación

4.3, obtiene la clase de la función que minimiza su valor, es decir, la instancia x se asignará a la clase que minize el valor de g_i. Los empates se rompen al azar, es decir, x se asignan al azar a una de las clases.

4.2 Clasificadores base

Una vez teniendo las representaciones del corpus podemos clasificar las instancias, los clasificadores utilizados para la experimentación en este trabajo son: Máquina de Soporte Vectorial, Naive Bayes y Árboles de Desición. A continuación se presenta una descripción de los clasificadores base.

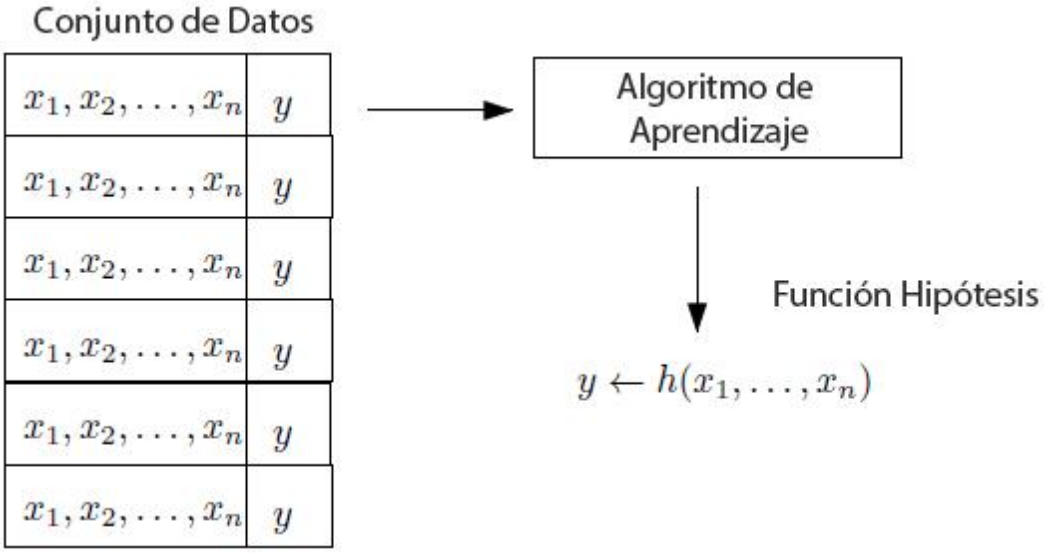

FIGURE 4.1: El conjunto de datos es dividido en n particiones que serán la entrada del algoritmo de aprendizaje, el cual utiliza una función hipótesis para llevar a cabo la clasificación

4.2.1 Naive Bayes

Es un clasificador probabilístico que aplica el Teorema de Bayes para estimar la probabilidad posterior P(y | x) de la clase y dada la variable x

$$P(y|x) = \frac{P(y|x)P(y)}{P(x)} \tag{4.4}$$

Naive Bayes se centra en las probabilidades P(x|y) que se refieren a la verosimilitud y representan la probabilidad de observar el valor x, dado el valor de clase y. Debido a esto Naive Nayes es considerado un *clasificador generativo.*

De la ecuación 4.7 podemos observar que el denominador *P(x)* es constante para algún valor de y, por lo que no es necesario calcularlo en orden de tomar una predicción. Por lo tanto podemos usar la siguiente aproximación:

$$P(y|x) \ P(y|x)P(y) \tag{4.5}$$

El valor *P(y)* se refiere a la probabilidad anterior y puede ser estimada directamente por los datos. Sin embargo P(x|y) depende de la distribución conjunta de x dado y . Y dado que x es una variable aleatoria multivariable, P(x|y) es muy caro de estimar.

De acuerdo a la regla de la cadena, la distribución conjunta de P(x|y) puede ser expresada de la siguiente manera:

$$P(x_1, ..., x_n|y) = P(x_1|y)P(x_2|x_1, y)...P(x_n|x_{n-1}, .., x_2, x_1, y) \tag{4.6}$$

A manera de evitar la cara estimación de P(x|y) , el clasificador considera una fuerte suposición, que todos los pares de características x_i y x_j son independientes para cada evidencia de y dada. De esta manera se tiene *P(x_i|x_j,y)=P(x_i |y)* para algún par *i,j* ε *[1,n].* Por lo tanto la función de verosimilitud puede ser representada de acuerdo a la siguiente expresión:

$$P(x|y) = P(x_1|y)P(x_2|y)...P(x_n|y) = \prod_{i=1}^{n} P(x_i|y) \tag{4.7}$$

De esta manera las probabilidades P(x_i|y) pueden ser estimadas directamente de los datos.

4.2.2 Máquina de Soporte Vectorial

La máquina de Soporte Vectorial (SVM) es un clasificador binario discriminante, dirigido a encontrar el hiperplano óptimo ($w^T * x + b$) que separa los dos posibles valores de la

variable etiquetada y ε {*+1,-1*} de acuerdo al espacio de características representado por x. El hiperplano óptimo es aquel que maximiza el margen entre las instancias positivas y negativas en el conjunto de datos de entrenamiento formado por N observaciones. La tarea de aprendizaje de una SVM se formaliza con el siguiente problema de optimización:

$$\min_{w,b} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N} \xi_i \tag{4.8}$$

$$\text{sujeto a } \; y_i(w^T x_i + b) \geq 1 - \xi_i \; \forall i\varepsilon\{1, ...N\}$$

$$\xi_i \geq 0, \; \forall i\varepsilon\{1, .., N\}$$

El objetivo del problema se enfoca en dos aspectos, el primero, obtener el máximo margen en el hiperplano y minimizar el error $\sum_i^N xi_i$. El parámetro C se refiere al parámetro suave de regularización de margen y controla la sensibilidad de la SVM para los posibles valores atípicos.

También es posible hacer que las SVM encuentren patrones no lineales, de manera eficiente usando el kernel *trick*. Una función $\phi(x)$ que mapea el espacio de características x usando un espacio dimensional alto conocido como el *espacio de Hilbert*, donde el producto punto $\phi(x)\phi(x')$ es conocido como la función kernel *K(x,x')*. De esta manera, el hiperplano es calculado en un espacio de dimensión alta $w^T\phi(x) + b$. La formulación dual de las SVM permite reemplazar cada producto punto por una función kernel como se muestra en la siguiente expresión:

$$\max_{\alpha} \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{N} \alpha_i\alpha_j y_i y_j K(x_i, x_j) \tag{4.9}$$

$$\text{sujeto a } \; \alpha_i \geq 0, \; \forall i\varepsilon\{1, .., N\}, \; \sum_{1=1}^{N} \alpha_i y_i = 0$$

Donde los parámetros α_i, i $\varepsilon\{1, ..., N\}$ corresponde a los *multiplicadores de Lagrange* del problema de optimización. Una vez que los parámetros se determinaron, es posible clasificar nuevas observaciones x_j de acuerdo a la siguiente expresión.

$$sign\left(\sum_{i=1}^{N} \alpha_i y_i K(x_i, x_j) + b\right) \tag{4.10}$$

4.2.3 El algoritmo de KNN

Del inglés *(NN, Nearest Neighbours)*, es una técnica que basa su criterio de aprendizaje en la hipótesis de que los miembros de una población suelen compartir propiedades y características con los individuos que los rodean, de modo que es posible obtener información descriptiva de un individuo mediante la observación de sus vecinos más cercanos, dicho algoritmo se ha convertido en uno de los métodos de clasificación más usados. Se define de la siguiente manera:

Sean c clases $w_1, w_2, ..., w_c$ dentro de un problema de clasificación de documentos, con muestas desconocidas de los k vecinos cercanos en donde la mayoría de los k vecinos pertenecen a la misma clase que llamaremos X. Supongamos que $k_1, k_2, ..., k_c$ es el número de muestras que pertenecen a las clases $w_1, w_2, ..., w_c$ en los k vecinos cercanos. La función de decisión se define como:

$$\mu_i(X) = k_i, \quad i = 1, 2, ..., c \tag{4.11}$$

De acuerdo a esta fórmula, la regla de decisión es:

$$if\ \mu_j(X) = \max_i k_i\ then\ X\ \epsilon\ w_j \tag{4.12}$$

Dicha regla es llamada DVF (función de valores discretos). Sin embargo en la actualidad, la regla de decisión SWF (Función de Similitud de pesos) [13], es ampliamente utilizada en los sistemas de clasificación de texto kNN. Los sistemas buscan los k documentos (llamados vecinos), los cuales tienen la máxima similitud para con otros vecinos, en un conjunto de entrenamiento. De acuerdo a las similitudes los vecinos son afiliados a cierta clase. Las similitudes entre los documentos vecinos y los documentos de prueba pueden ser calculadas de la siguiente manera:

$$\mu_j(X) = \sum_{i=1}^{k} \mu_i(X_i) sim(X, X_i) \tag{4.13}$$

Donde $\mu_j(X_i)\epsilon\{0,1\})$ muestra si X_i pertenece a w_j $\mu_j(X_i) = 1$ es verdadera y si $w_j\left(\mu_j(X_i)\right) = 0$ si es falso, $sim(X, X_i)$ denota la similitud entre los documentos de entrenamiento y los documentos de prueba [5].

4.2.4 Árboles de Decisión

Un árbol de desición describe un conjunto de reglas organizado de forma jerárquica, que implementan una estructura de decisión. Se compone de hojas y nodos. Una hoja registra una respuesta *(clase)* y un nodo especifica algunas condiciones de las pruebas que se llevarán a cabo en un valor único, rasgo de una instancia, con una rama y sub árbol para cada posible resultado de la prueba. Para un determinada vector , se toma una decisión partiendo de la raíz de un árbol y se mueve a través del árbol determinado por el resultado de una prueba de estado en cada nodo hasta que se encuentra una hoja de [18]. El proceso de construcción de un árbol de decisión es una partición recursiva de un conjunto de entrenamiento.

A continuación se listan algunas de sus características

- Si todos los objetos son distinguibles, entonces podemos construir un clasificador árbol con error de resubstitution cero. Este hecho permite que el árbol sea capaz de memorizar los datos de entrenamiento para que pequeñas alteraciones pudieran conducir a un clasificador árbol estructurado de manera diferente. La inestabilidad puede ser una ventaja más que un inconveniente cuando se consideran los conjuntos de clasificadores
- Los clasificadores de árboles son intuitivos porque el proceso de decisión puede ser rastreado como una secuencia de decisiones simples.
- Para dicho método son adecuadas las características cuantitativas y cualitativas, Con un pequeño número de categorías son especialmente útiles porque la decisión puede ser fácilmente diversificada. Para características cuantitativas, un punto de división tiene que ser encontrado para transformar la función en datos categóricos. Los árboles de decisión no se basan en un concepto de distancia en el espacio de características. Son principalmente útiles cuando se tienen cararterísticas categóricas o mixtas. Esta es la razón pos la cual los árboles de decisión se consideran como métodos no métricos para la clasificación.

4.3 Ensamble de clasificadores

La primera etapa para construir un ensamble de clasificadores involucra el proceso de generación de una colección de clasificadores base, un enfoque es aplicar N diferentes métodos de aprendizaje, con un solo conjunto de datos de entrenamiento, para obtener N diferentes modelos de clasificación [19].

Otro enfoque es crear N diferentes particiones de los datos de entrenamiento y emplear un solo algoritmo de aprendizaje con cada partición [20]. El principal problema en este enfoque es la conversión del conjunto de datos originales en una colección de diferentes conjuntos de datos de entrenamiento. En algunas técnicas, el conjunto de datos original está dividido en N sub conjuntos seleccionados aleatoriamente. Otros trabajos involucran la manipulación de los datos de acuerdo a la distribución de los datos.

Dado el potencial uso del ensamble de clasificadores, existen algunos factores que deben ser diferenciados entre los varios métodos de ensamble. Los principales factores se listan a continuación:

1. Relación inter-clasificadores. ¿Cómo cada clasificador afecta a otros clasificadores? Los métodos de ensamble pueden ser divididos en dos principales tipos: secuenciales y concurrentes.

2. Método de combinación. La estrategia de combinar los clasificadores generados por un algoritmo de inducción. El combinador simple determina la salida exclusivamente a partir de las salidas de los inductores individuales.

3. Generador de diversidad. Con el objetivo de realizar un ensamble eficiente, debe existir diversidad entre los clasificadores involucrados. La diversidad puede ser obtenida a través de diferentes presentaciones de entrada de datos, como en bagging, variaciones en el diseño de aprendizaje, aplicando una sanción a las salidas para fomentar la diversidad.

A continuación se describen las técnicas más populares de ensamble de clasificadores.

4.3.1 Cascada

También conocida como generalización de cascada, es una arquitectura para combinar clasificadores como se muestra en la figura 2, puede presentar n niveles, sin embargo normalmente presenta dos niveles, en donde el nivel 1 es entrenado con el conjunto de datos original, el nivel 2 con un conjunto de datos aumentado, el cual contiene las características del conjunto de datos original junto con la salida del clasificador del nivel 1. La salida del clasificador del nivel 1 es un vector que contiene la distribución de probabilidad condicional $(p_1, ..., p_c)$, donde c es el número de clases del conjunto de datos original, y p_i es la estimación de probabilidad calculada por el clasificador del nivel 1 de que la instancia pertenezca a la clase i.

El entrenamiento del clasificador del nivel 2 es influenciado por el clasificador del nivel anterior, debido a que considera su salida obtenida, derivando un esquema global sobreentrenado. Sin embargo, en cascada se reduce este problema debido a dos razones: en cada nivel se utiliza un clasificador de diferente naturaleza a del otro y además el clasificador del nivel 2 no se entrena únicamente con la salida del clasificador de nivel 1, sino que además tiene en cuenta las características originales.

Cascada combina dos clasificadores, seleccionando aquel clasificador que obtenga un error bajo de bias, y otro con valor de varianza baja también, para conseguir uno nuevo que tenga valores bajos en ambas medidas. En el trabajo realizado en [21] se ocupa el clasificador con error de varianza baja en el nivel 1 y el clasificador con error de bias baja en el nivel 2, debido a que seleccionando métodos con bajo bias en el nivel superior, es posible ajustarse a áreas de decisión más complejas, teniendo en cuenta las superficies estables, trazadas por el clasificador o los clasificadores de nivel inferior. En [21], se realiza una validación experimental utilizando 26 conjuntos de datos del repositorio de la universidad de California, Irvine *(University of California, Irvine UCI)* que da soporte a realizar el ensamble de clasificadores de ésta manera.

4.3.2 Mayoría de votos

Es un método simple de combinación de clasificadores base, en el cual todos los clasificadores incluidos proveen un voto a alguna de las clases, el método realiza la sumatoria

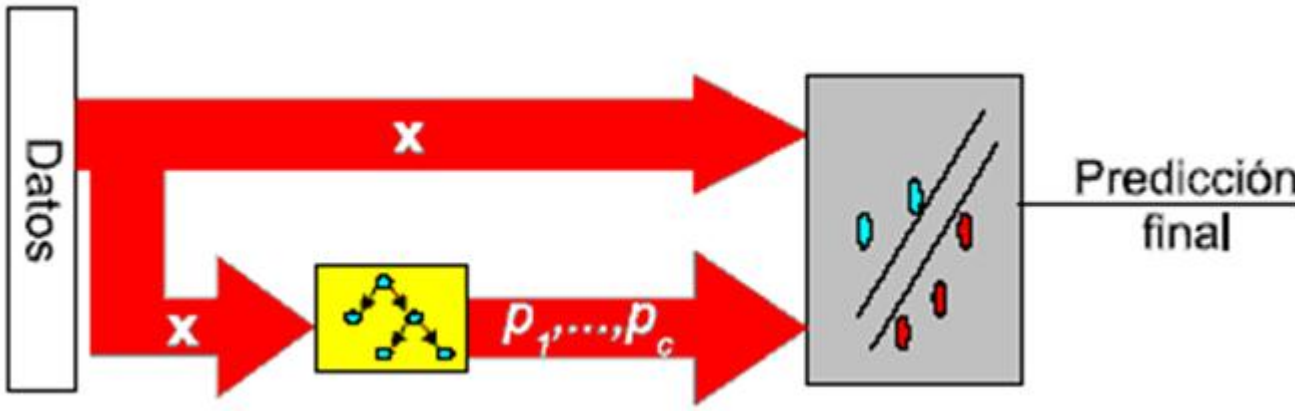

FIGURE 4.2: Estructura de Cascada de 2 niveles

de dichos votos y la clase que recibe más votos es seleccionada como la decisión final. Dicho método es representado por la siguiente ecuación:

$$x \rightarrow w \; if \; w = \arg\max_{w \epsilon \theta} \sum_{i=1}^{T} 1(C_i(x) = w) \tag{4.14}$$

$$1(C_i(x) = w) = \begin{cases} 1 & \text{si } C_i(x) = w \\ 0 & \textit{en otro caso} \end{cases} \tag{4.15}$$

x es una instancia, θ es el conjunto de etiquetas de clase, w es la clase asignada para la instancia x y C_1,...,C_T son los clasificadores base.

Este método a pesar de ser tan simple ha sido ampliamente utilizado en diferentes áreas, un ejemplo de ello se describe en [22] donde es utilizado junto con la transformada de *"Haar"* para mejorar la efectividad de los sistemas de autenticación basados en reconocimiento de iris. Otro trabajo más donde también se utiliza el esquema de mayoría de votos se describe en [23] en el cual dicho método se utiliza para detectar canales cubiertos maliciosamente dentro de una red a través de un túnel DNS.

Así como el esquema de mayoría de votos ha tenido éxito en otras áreas de conocimiento, se utiliza este método al problema planteado.

4.3.3 Ventanas

El método de *Ventanas* es una técnica general, que tiene por objetivo mejorar la eficiencia de los métodos de aprendizaje o clasificadores utilizados, mediante la identificación de un subconjunto adecuado de instancias de entrenamiento. Dicho método se lleva a cabo mediante el uso de un procedimiento de sub-muestreo.

El método funciona de la siguiente manera: Se selecciona un subconjunto aleatorio de instancias para el entrenamiento de un clasificador *(una ventana)*, el resto de instancias son utilizadas para los datos de prueba, si la precisión obtenida del clasificador es insuficiente, las instancias de prueba clasificadas erróneamente se eliminan de las instancias de prueba y se añaden al conjunto de instancias para el entrenamiento en la siguiente iteración. En 1993 [24] Quinlan propone dos formas diferentes de la formación de una ventana: en la primera, la ventana actual se extiende hasta un límite especificado. En la segunda, varios casos "clave" en la ventana actual se identifican y el resto son reemplazados. Así, el tamaño de la ventana se mantiene constante. El proceso continúa hasta que se obtiene una precisión suficiente, y el clasificador construido a la última iteración es elegido como el clasificador final.

El método de ventanas tambien ha sido estudiado por Fürnkranz en 1997, [25] en donde se muestra que para este tipo de algoritmo, se pueden presentar mejoras significativas en la eficiencia solo en dominios libres de ruido. En dicho trabajo se propone un versión de ventanas en donde se elimina de los datos de entrenamiento todos los casos que han sido clasificados correctamente, y agregan todos los casos que han sido clasificados erróneamente. La eliminación de instancias desde la ventana mantiene su pequeño tamaño y por lo tanto disminuye el tiempo de ejecución.

En conclusión, en ambos casos el método de ventanas construye una secuencia de clasificadores para obtener una muestra final. Es importante mencionar que ventanas no combina clasificadores, su tarea radica en mejorar el resultado de un clasificador.

Una vez que se ha explicado los métodos de clasificación y arquitecturas utilizadas, es importante conocer las métricas que permitirán evaluar el resultado obtenido por los mismos. A continuación se presentan las métricas de evaluación utilizadas.

	y=+1	y=-1
c(x)=+1	TP	FP
c(x)=-1	FN	TN

TABLE 4.1: Combinación de clasificaciones

4.3.4 Métricas de evaluación

Para realizar la evaluación de los métodos de clasificación aplicados sobre un *dataset*, se describen a continuación las métricas utilizadas.

Dado que el problema planteado esta formulado como un problema de clasificación binaria, se definen los términos utilizados por las métricas en la tabla 4.1.

Donde c es la clasificación asignada al valor x, y y es la clase correcta de la instancia, TP son las instancias clasificadas correctamente como positivas, *FP*, son las instancias clasificadas erróneamente como positivas y de la misma manera para las instancias negativas, *FN*, son las instancias clasificadas erróneamente como positivas y *TN* son las clasificadas correctamente como negativas. Ahora teniendo las salidas antes descritas los siguientes criterios de evaluación pueden ser utilizados.

Precisión. Es la fracción de observaciones clasificadas correctamente como positivas, sobre todas las observaciones clasificadas como positivas.

$$Precision = \frac{TP}{TP + FP} \tag{4.16}$$

Recuerdo. Es la fracción de observaciones clasificadas correctamente como positivas, sobre todas las observaciones positivas.

$$Recuerdo = \frac{TP}{TP + FN} \tag{4.17}$$

Medida F. Es el significado armónico entre precisión y recuerdo

$$MedidaF = \frac{(1 + \beta^2)(2 * Presicion * Recuerdo)}{(\beta^2 * Precision) + Recuerdo} \tag{4.18}$$

Las medidas de evaluación son promediadas por todas las sub muestras, asegurando que todas las observaciones fueron usadas para entrenamiento y prueba.

Chapter 5

Experimentos y Resultados

En este capítulo se presentan los experimentos realizados, los métodos de clasificación utilizados son: Máquina de Soporte Vectorial (SVM), Árboles de decisión y Naive Bayes. Y los metodos de combinación de clasificadores son Mayoría de votos, Cascada y Ventanas.

5.1 Corpus utilizados

Para la realización de los experimentos se consideraron 3 corpus, uno en el idioma español y 2 de idioma inglés. A continuación se describen

- **Corpus Español**

  Es un corpus de opiniones de películas de cine, creado en [9], con 3878 críticas que contienen una puntuación asignada del 1 al 5 donde 1 es la más negativa y 5 es la más positiva, del cual se tomaron 2625 críticas (1351 positivas, 1274 negativas) no incluyendo las criticas neutras es decir con puntuación 3.

- **Corpus en idioma Inglés con 2000 opiniones**

  El segundo corpus es de opiniones de cine en inglés, con 1000 opiniones negativas y 1000 opiniones positivas, extraidas de la página: http://www.rottentomatoes.com/

- **Corpus en idioma Inglés con 10662**

Es un corpus de críticas de cine con 10662 opiniones, 5331 positivas y 5331 negativas, extraidas tambien de la página http://www.rottentomatoes.com/

5.2 Pre Procesamiento de los datos

Una vez elegidos los corpus y antes de realizar el Análisis de Sentimientos, primero se debe realizar un pre procesamiento de los datos, ya que los corpus fueron construidos a partir de opiniones introducidas por usuarios comunes de la web y no por críticos especializados, lo cual dificulta la tarea del procesamiento de los datos, pues el corpus puede contener palabras vacías, faltas de ortografía, incoherencias, palabras incompletas, etc.

A continuación se lista las acciones realizadas en el pre procesamiento de los datos, para los 3 corpus utilizados.

1. Eliminación de símbolos no alfanuméricos.
    - Eliminación de símbolos de puntuación.
    - Eliminación de números.
    - Eliminación de palabras vacías.

5.3 Condiciones de ejecución

Los experimentos fueron realizados en Matlab 2014a. Los experimentos realizados fueron variando el porcentaje de datos de entrenamiento y de prueba, con 80% - 20% y 60%-40% respectivamente.

5.4 Experimentos

Corpus	Representación	Clasificador	Precisión	Recuerdo	Medida F
Español	Bigramas	SVM	0.824138	0.821306	0.822739
		arboles	0.646429	0.621993	0.633995
		Naive Bayes	0.563953	1	0.72121
		Mayoria de votos	0.687179	0.920962	0.787098
		Cascada s-s	0.856655	0.865517	0.861083
		Ventana s-s-s-s	0.978799	0.965157	0.97195
	tf-idf	SVM	0.683824	0.319588	0.435617
		arboles	0.557823	0.563574	0.560704
		Naive Bayes	0.655462	0.536082	0.589812
		Mayoria de votos	0.655	0.450172	0.533625
		Cascada s-s	0.572519	0.570342	0.571449
		Ventana s-s-s-s	0.952569	0.919847	0.935942
	POS	SVM	0.580205	0.584192	0.582212
		arboles	0.586614	0.512027	0.546809
		Naive Bayes	0.579882	0.67354	0.623231
		Mayoria de votos	0.592233	0.628866	0.61002
		Cascada s-s	0.536184	0.619772	0.574976
		Ventana s-s-s-s	0.922118	0.945687	0.933774
	Valoración	SVM	0.75	0.597938	0.665412
		arboles	0.658915	0.584192	0.619328
		Naive Bayes	0.733945	0.549828	0.628704
		Mayoria de votos	0.748899	0.584192	0.656391
		Cascada s-s	0.655462	0.614173	0.634166
		Ventana s-s-s-s	0.982833	0.938525	0.960188
		Mayoria de votos Mejores	0.995902	0.972	0.983826
Inglés 2000	Bigramas	SVM	0.846154	0.858537	0.85232
		arboles	1	1	1
		Naive Bayes	0.508861	0.980488	0.67002
		Mayoria de votos	0.86383	0.990244	0.922747
		Cascada s-s	1	1	1
		Ventana s-s-s-s	0.995122	1	0.997575
	tf-idf	SVM	1	1	1
		arboles	1	1	1
		Naive Bayes	0.990338	1	0.995166
		Mayoria de votos	1	1	1
		Cascada s-s	1	1	1
		Ventana s-s-s-s	0.995122	1	0.997575
	POS	SVM	0.995098	0.990244	0.992685
		arboles	1	1	1
		Naive Bayes	0.990196	0.985366	0.987795
		Mayoria de votos	0.995146	1	0.997587
		Cascada s-s	1	1	1
		Ventana s-s-s-s	0.995122	1	0.997575
		Mayoria de votos Mejores	1	1	1
Inglés 10662	Bigramas	SVM	0.989681	0.970561	0.980048
		arboles	1	1	1
		Naive Bayes	0.50985	1	0.675385
		Mayoria de votos	0.989982	1	0.994986
		Cascada s-s	0.998162	0.99908	0.998641
		Ventana s-s-s-s	0.989748	0.981516	0.985635
	tf-idf	SVM	1	1	1
		arboles	1	1	1
		Naive Bayes	0.998163	1	0.999101
		Mayoria de votos	1	1	1
		Cascada s-s	1	0.824595	0.903886
		Ventana s-s-s-s	1	1	1
	POS	SVM	0.998162	0.99908	0.998641
		arboles	1	1	1
		Naive Bayes	0.998111	0.972401	0.985109
		Mayoria de votos	0.998162	0.99908	1
		Cascada s-s	0.998162	1	0.9991
		Ventana s-s-s-s	1	1	1
		Mayoria de votos Mejores	1	1	1

Table 5.1: Tabla Inglés 80%-20%

Corpus	Representación	Clasificador	Precisión	Recuerdo	Medida F
		SVM	0.715736	0.776147	0.744738
		arboles	0.613508	0.6	0.606699
Español	Bigramas	Naive Bayes	0.830769	0.099083	0.177069
		Mayoria de votos	0.775676	0.526606	0.627342
		Cascada s-s	0.798903	0.800366	0.799654
		Ventana s-s-s-s	0.809107	0.838475	0.823549
		SVM	0.647482	0.330275	0.437444
		arboles	0.536101	0.544954	0.540511
Español	tf-idf	Naive Bayes	0.605691	0.546789	0.574755
		Mayoria de votos	0.635697	0.477064	0.545093
		Cascada s-s	0.528169	0.549451	0.53862
		Ventana s-s-s-s	0.968468	0.617816	0.754406
		SVM	0.578348	0.372477	0.453145
		arboles	0.575875	0.543119	0.559038
Español	POS	Naive Bayes	0.571942	0.291743	0.386411
		Mayoria de votos	0.59816	0.357798	0.447781
		Cascada s-s	0.522807	0.545788	0.53407
		Ventana s-s-s-s	0.894558	0.477314	0.622505
		SVM	0.693182	0.559633	0.619309
		arboles	0.598148	0.592661	0.595412
Español	Valoración	Naive Bayes	0.666667	0.502752	0.573242
		Mayoria de votos	0.681093	0.548624	0.607744
		Cascada s-s	0.538033	0.531136	0.534582
		Ventana s-s-s-s	0.863222	0.704715	0.775976
		Mayoria de votos Mejores	0.732314	0.702752	0.717248
		SVM	0.785219	0.862944	0.822269
		arboles	1	1	1
Inglés 2000	Bigramas	Naive Bayes	0.491206	0.992386	0.657163
		Mayoria de votos	0.8107	1	0.895475
		Cascada s-s	0.997561	1	0.998799
		Ventana s-s-s-s	1	1	1
		SVM	1	0.997462	0.998749
		arboles	1	1	1
Inglés 2000	tf-idf	Naive Bayes	0.994924	0.994924	0.994944
		Mayoria de votos	1	0.997462	0.998749
		Cascada s-s	0.997561	1	0.998799
		Ventana s-s-s-s	1	1	1
		SVM	1	0.994924	0.997475
		arboles	1	1	1
Inglés 2000	POS	Naive Bayes	1	0.92132	0.959069
		Mayoria de votos	1	0.994924	0.997475
		Cascada s-s	0.997561	1	0.998799
		Ventana s-s-s-s	1	1	1
		Mayoria de votos Mejores	1	1	1
		SVM	0.987648	0.966078	0.976764
		arboles	1	1	1
Inglés 10662	Bigramas	Naive Bayes	0.50469	1	0.670843
		Mayoria de votos	0.988062	1	0.994015
		Cascada s-s	0.999522	1	0.999781
		Ventana s-s-s-s	1	1	1
		SVM	1	1	1
		arboles	1	1	1
Inglés 10662	tf-idf	Naive Bayes	1	0.998141	0.99909
		Mayoria de votos	1	1	1
		Cascada s-s	0.999522	1	0.999781
		Ventana s-s-s-s	1	1	1
		SVM	0.999071	1	0.999556
		arboles	1	1	1
Inglés 10662	POS	Naive Bayes	0.999039	0.966543	0.982542
		Mayoria de votos	0.999071	1	0.999556
		Cascada s-s	0.999522	1	0.999781
		Ventana s-s-s-s	1	1	1
		Mayoria de votos Mejores	1	1	1

Table 5.2: Tabla Inglés 60%-40%

Chapter 6

Conclusiones

Bibliography

[1] Web usage mining. In *Web Data Mining*, Data-Centric Systems and Applications, pages 449–483. Springer Berlin Heidelberg, 2007. ISBN 978-3-540-37881-5. doi: 10.1007/978-3-540-37882-2_12. URL http://dx.doi.org/10.1007/978-3-540-37882-2_12.

[2] Bing Liu 0001 and Lei Zhang 0016. A survey of opinion mining and sentiment analysis. In Charu C. Aggarwal and ChengXiang Zhai, editors, *Mining Text Data*, pages 415–463. Springer, 2012. ISBN 978-1-4419-8462-3. URL http://dblp.uni-trier.de/db/books/collections/Mining2012.html#LiuZ12.

[3] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of EMNLP*, pages 79–86, 2002.

[4] Bing Liu, Bamshad Mobasher, and Olfa Nasraoui. Web usage mining. In *Web Data Mining*, Data-Centric Systems and Applications, pages 527–603. Springer Berlin Heidelberg, 2011. ISBN 978-3-642-19459-7. doi: 10.1007/978-3-642-19460-3_12. URL http://dx.doi.org/10.1007/978-3-642-19460-3_12.

[5] Wenqian Shang, Houkuan Huang, Haibin Zhu, Yongmin Lin, Zhihai Wang, and Youli Qu. An improved knn algorithm – fuzzy knn. In Yue Hao, Jiming Liu, Yuping Wang, Yiu-ming Cheung, Hujun Yin, Licheng Jiao, Jianfeng Ma, and Yong-Chang Jiao, editors, *Computational Intelligence and Security*, volume 3801 of *Lecture Notes in Computer Science*, pages 741–746. Springer Berlin Heidelberg, 2005. ISBN 978-3-540-30818-8. doi: 10.1007/11596448_109. URL http://dx.doi.org/10.1007/11596448_109.

[6] Ian H. Witten, Eibe Frank, and Mark A. Hall. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 3rd edition, 2011. ISBN 0123748569, 9780123748560.

[7] Peter D. Turney. Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 417–424, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073153. URL http://dx.doi.org/10.3115/1073083.1073153.

[8] Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. Lexicon-based methods for sentiment analysis. *Comput. Linguist.*, 37(2):267–307, June 2011. ISSN 0891-2017. doi: 10.1162/COLI_a_00049. URL http://dx.doi.org/10.1162/COLI_a_00049.

[9] Bing Liu. Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1):1–167, 2012. doi: 10.2200/S00416ED1V01Y201204HLT016. URL http://dx.doi.org/10.2200/S00416ED1V01Y201204HLT016.

[10] Michael John Collins. A new statistical parser based on bigram lexical dependencies. In *Proceedings of the 34th Annual Meeting on Association for Computational Linguistics*, ACL '96, pages 184–191, Stroudsburg, PA, USA, 1996. Association for Computational Linguistics. doi: 10.3115/981863.981888. URL http://dx.doi.org/10.3115/981863.981888.

[11] F. Benamara, C. Cesarano, A. Picariello, D. Reforgiato, and V.S. Subrahmanian. Sentiment analysis: Adjectives and adverbs are better than adjectives alone. In *Proceedings of the International Conference on Weblogs and Social Media (ICWSM)*, 2007.

[12] Ellen Riloff, Janyce Wiebe, and Theresa Wilson. Learning subjective nouns using extraction pattern bootstrapping. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, pages 25–32, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics. doi: 10.3115/1119176.1119180. URL http://dx.doi.org/10.3115/1119176.1119180.

[13] Peter R. R. White. Appraisal outline. Stroudsburg, PA, USA, 2001. Association for Computational Linguistics. URL www.grammatics.com/appraisal.

[14] V. M. Morales de Jesús. Utilización de expresiones de actitud para el análisis de sentimientos. Puebla, Puebla., 2014.

[15] Kai Ming Ting and Ian H. Witten. Stacked generalization: when does it work? In *in Procs. International Joint Conference on Artificial Intelligence*, pages 866–871. Morgan Kaufmann, 1997.

[16] Kagan Tumer and Joydeep Ghosh. Linear and order statistics combiners for pattern classification. *CoRR*, cs.NE/9905012, 1999. URL http://arxiv.org/abs/cs.NE/9905012.

[17] Richard O. Duda and Peter E. Hart. *Pattern Classification and Scene Analysis*. A Wiley Interscience Publication. Wiley, 1973.

[18] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern Classification*. Wiley, 2001.

[19] J. Kittler, M. Hatef, R.P.W. Duin, and J. Matas. On combining classifiers. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(3):226–239, Mar 1998. ISSN 0162-8828. doi: 10.1109/34.667881.

[20] Thomas G. Dietterich. Ensemble methods in machine learning. In *Proceedings of the First International Workshop on Multiple Classifier Systems*, MCS '00, pages 1–15, London, UK, UK, 2000. Springer-Verlag. ISBN 3-540-67704-6. URL http://dl.acm.org/citation.cfm?id=648054.743935.

[21] João Gama and Pavel Brazdil. Cascade generalization. *Machine Learning*, 41 (3):315–343, 2000. ISSN 0885-6125. doi: 10.1023/A:1007652114878. URL http://dx.doi.org/10.1023/A%3A1007652114878.

[22] V. Anitha and R.Leela Velusamy. Iris recognition systems with reduced storage and high accuracy using majority voting and haar transform. Advances in Intelligent Systems and Computing, pages 813–822. Springer Berlin Heidelberg, 2012. ISBN 978-3-642-30110-0. doi: 10.1007/978-3-642-30111-7_78. URL http://dx.doi.org/10.1007/978-3-642-30111-7_78.

[23] Maurizio Aiello, Maurizio Mongelli, and Gianluca Papaleo. Supervised learning approaches with majority voting for dns tunneling detection. In *International Joint*

Conference SOCO'14-CISIS'14-ICEUTE'14, volume 299 of *Advances in Intelligent Systems and Computing*, pages 463–472. Springer International Publishing, 2014. ISBN 978-3-319-07994-3. doi: 10.1007/978-3-319-07995-0_46. URL http://dx.doi.org/10.1007/978-3-319-07995-0_46.

[24] J. Ross Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993. ISBN 1-55860-238-0.

[25] Johannes Fürnkranz. More efficient windowing. Technical Report OEFAI-TR-97-01, Austrian Research Institute for Artificial Intelligence, Wien, Austria, 1997. URL http://www.ke.informatik.tu-darmstadt.de/~juffi/publications/aaai-97.ps.gz.